



**Mobility-Dependent Motion Planning
for High Speed Robotic Vehicles**

ARO Award Number: W911NF-07-1-049

Final Report

8/1/2007 – 7/31/2008

Prepared for:

U.S. Army Research Office
Systems and Control Division
P.O. Box 12211
4300 South Miami Blvd.
Research Triangle Park, NC 27709-2211

Attention:

Dr. Randy Zachery
Tel: (919) 549-4368, Fax: (919) 549-4354
Email: randy.zachery@us.army.mil

Technical POC:

Dr. Karl Iagnemma
Department of Mechanical Engineering
Massachusetts Institute of Technology
77 Massachusetts Avenue, Room 3-435a
Cambridge, MA 02139
Tel #: 617-452-3262 Fax #: 617-253-9637
Email: kdi@mit.edu

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 25 JUL 2008		2. REPORT TYPE		3. DATES COVERED 01-08-2007 to 31-07-2008	
4. TITLE AND SUBTITLE Mobility-Dependent Motion Planning for High Speed Robotic Vehicles				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Massachusetts Institute of Technology, Office of Sponsored Programs, Bldg E19-750, Cambridge, MA, 02139-4307				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 76	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

1 Foreword

The ultimate objective of this work is to design planning and control architectures that enable autonomous vehicles to operate in open terrain without sacrificing speed and maneuverability. To this end, we develop mobility metrics for UGVs operating off-road at high-speed regimes, explore optimal algorithms to derive optimal paths for UGVs, and propose a framework for motion planning. The major tasks to accomplish this goal are the following:

- Develop an efficient stochastic terrain traversal prediction tool that allows systematic assessment of the ability of a vehicle to negotiate challenging terrain. This tool will improve the computational efficiency of current Monte Carlo-based methods and allow for the construction of a large library of obstacle traversal maneuvers.
- Design vehicle mobility metrics that quantify the ability of a vehicle to traverse challenging terrain and to track trajectories with varying curvature and speed. These metrics will be unique since they will: a) represent a UGVs obstacle traversal capability as a function of its trajectory (i.e., path and velocity), and b) consider UGV agility (i.e., the ability to track high-curvature paths, and execute challenging maneuvers).
- Develop methods for evaluation of terrain/control strategy pairs that provide a compact yet accurate way to identify, classify, and assess terrain-specific maneuvers. This task will bridge the gap between accurate but complex metrics, and tractable metrics that can be used in real-time path planning.
- Develop a symbolic control framework that permits the execution of complex, high-performance trajectories through the on-line combination of a number of baseline controllers, chosen from a finite library of elementary behaviors or maneuvers.

2 Table of Contents

1	Foreword	1
2	Table of Contents	2
3	Problem Statement	3
4	Summary of Results	6
4.1	Efficient stochastic terrain traversal.	6
4.2	Optimal motion planning on heterogeneous terrain.	9
4.3	Robust control algorithms for motion primitive selection. . . .	11
A	Appendix: Mobility Prediction for Unmanned Ground Vehicles in Un- certain Environments	14
B	Appendix: On the Optimality of Dubins Paths across Heterogeneous Terrain	27
C	Appendix: A Hybrid Control Framework for Robust Maneuver-Based Motion Planning	47
D	Appendix: Discussion on “Optimality Properties and Driver Input Pa- rameterization for Trail-Braking Cornering”	66

3 Problem Statement

In this project, we consider the problem of autonomous driving of land vehicles in open terrain at high speeds. Achieving autonomous operation of these vehicles in open terrain remains a challenging problem not only due to uncertainty of the operating environment but also due to the limited knowledge of the vehicle’s mobility in such type of terrain. Moreover, autonomous driving of wheeled vehicles at high speeds adds a new level of complexity due to the time constraints imposed by the small reaction times and the nonlinear characteristics of the vehicle’s operation. A typical scenario for autonomous driving in open terrain is depicted in Figure 1, where an Unmanned Ground Vehicle (UGV) deploys across a rocky, sloped, and vegetation-covered environment.



Figure 1: Autonomous driving of an Unmanned Ground Vehicle (UGV) across an open terrain with rocks, slopes, and vegetation.

High-speed autonomous navigation in rough terrain, like the scenario in Figure 1, is challenging because navigation algorithms must consider nonlinear vehicle dynamic effects such as wheel slip, skidding, ballistic behavior, rollover, and vehicle-terrain interaction phenomena. Navigation algorithms must also consider the presence of obstacles. Additionally, these algorithms

must be computationally efficient enough to operate in real time. Further difficulties arise due to real-world uncertainties caused by unknown/time-varying vehicle parameters, unknown/poorly known terrain conditions, and range sensor error and uncertainty. The reason for the poor understanding of mobility in open terrain, that is, its ability to traverse rugged terrain efficiently and safely, is rooted in the intrinsic complexity of the vehicle’s dynamics and in the vehicle/terrain interaction. These properties are hard to quantify with a single scalar metric involving vehicle and terrain features. Generally speaking, a solution to the motion planning problem for high-speed autonomous vehicles in a highly unstructured and uncertain environment requires not only sophisticated vehicle design and hardware components (e.g., for actuation, sensing, and communication), but most importantly, it demands advanced software algorithms and supervisory control strategies that can make use of the full capabilities of these components.

The ultimate objective of this work is to design planning and control architectures that enable autonomous vehicles to operate in open terrain without sacrificing speed and maneuverability. To this end, we develop mobility metrics for UGVs operating off-road at high-speed regimes, explore optimal algorithms to derive optimal paths for UGVs, and propose a framework for motion planning. The major tasks to accomplish this goal are the following:

- Develop an efficient stochastic terrain traversal prediction tool that allows systematic assessment of the ability of a vehicle to negotiate challenging terrain. This tool will improve the computational efficiency of current Monte Carlo-based methods and allow for the construction of a large library of obstacle traversal maneuvers.
- Design vehicle mobility metrics that quantify the ability of a vehicle to traverse challenging terrain and to track trajectories with varying curvature and speed. These metrics will be unique since they will: a) represent a UGVs obstacle traversal capability as a function of its trajectory (i.e., path and velocity), and b) consider UGV agility (i.e., the ability to track high-curvature paths, and execute challenging maneuvers).
- Develop methods for evaluation of terrain/control strategy pairs that provide a compact yet accurate way to identify, classify, and assess

terrain-specific maneuvers. This task will bridge the gap between accurate but complex metrics, and tractable metrics that can be used in real-time path planning.

- Develop a symbolic control framework that permits the execution of complex, high-performance trajectories through the on-line combination of a number of baseline controllers, chosen from a finite library of elementary behaviors or maneuvers.

The results obtained from a research period of one year are summarized in the next section. A detailed presentation of these results as well as a thorough description of the methodologies employed to derive them can be found in Appendix A-B.

4 Summary of Results

4.1 Efficient stochastic terrain traversal.

We have developed a computationally efficient method for mobile robot mobility prediction. Such a method would be useful in analysis of high speed motion, especially in uneven, natural terrain. It would also find application in both motion planning and control algorithms designed for high speed scenarios.

Our approach to mobility prediction is based on the stochastic response surface method (SRSF). The SRSF is a method for efficient representation of the response of systems that are subject to uncertainty. In this approach, we represent model inputs as functions of normal random variables, with each having zero mean and unit variance. The same set of random variables is then used to represent a user-specified output. An equivalent reduced model for the output is expressed in the form of a series expansion, consisting of multi-dimensional Hermite polynomials of normal random variables, as:

$$y = a_0 + \sum_{i_1=1}^n a_{i_1} \Gamma_1(\xi_{i_1}) + \sum_{i_1=1}^n \sum_{i_2=1}^{i_1} a_{i_1 i_2} \Gamma_2(\xi_{i_1}, \xi_{i_2}) + \dots$$

where y refers to an output metric, a_{i_1}, a_{i_2}, \dots are coefficients to be determined, $\xi_{i_1}, \xi_{i_2}, \dots$ are i.i.d. normal random variables, and $\Gamma_q(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_q})$ is the Hermite polynomial of degree q .

The unknown coefficients are then estimated from a small number of model simulations, by choosing a set of sample points (i.e. collocation points), calculating the model output at these points, then applying regression to find the coefficients. The approximate reduced model was then used to analyze the system subject to uncertainty. It has been shown that this approach yields results that are comparable in accuracy to the outputs of classical Monte Carlo approaches.

The SRSF approach can be applied to a variety of applications such as those related to mobility prediction (including obstacle traversal, slope traversal, and rollover analysis) and motion planning (i.e. terrain dependent path generation) while explicitly considering uncertainty in terrain and/or vehicle parameters. The form of the uncertainty distribution can be specified, and can be state or position dependent.

For mobility analysis of a vehicle traversing sloped, deformable terrain, a simple description of mobility was defined as the probability that for a given

initial velocity at an initial position, the robot will have a non-negative velocity after moving up the incline. Taking into account uncertainty in terrain parameters, this was presented as a distribution of traversal probability versus initial velocity, which could then be used to predict for which velocities the robot will be able to traverse the deformable terrain region with a reasonably high probability. A classical Bekker-type wheel soil interaction model was used to calculate the drawbar pull (i.e. net longitudinal wheel thrust). An equivalent model for the drawbar pull was then formulated using the SRSM approach which was then used to calculate the traversal probabilities.

The approach was also used for mobility analysis of a vehicle traveling along a side slope. A probabilistic reachability metric was generated based on the statistics of the (spatial) distribution of trajectories resulting from a given open loop input over various distinct terrain types, and the mean trajectory was plotted along with the probability ellipsoids; the latter representing the deviation from the mean path under uncertainty. This is shown in Figure 2.

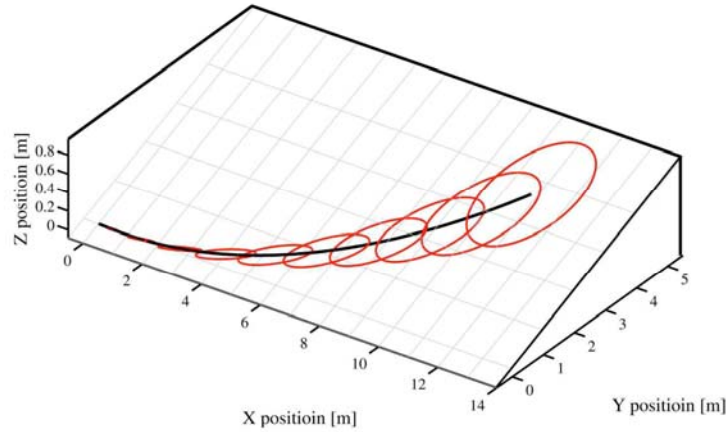


Figure 2: Mobility analysis “Reachability Metric” for vehicle traveling along a side-slope.

The SRSM method was also applied to rollover analysis of a Dubins vehicle for various steering high speed maneuvers. The rollover tendency was analyzed while considering uncertainty in the vehicle roll stiffness parame-

ters. The linear model considered included lateral acceleration, yaw and roll dynamics. It was shown that, as expected, explicit consideration of uncertainty yielded a richer description of the rollover probability as compared to a deterministic model-based approach.

Finally, initial investigation into integrating the SRSM approach with a motion planning method was performed. The motion planning method was based on rapidly exploring random trees (RRTs). Here, the SRSM method was used to calculate mobility-related statistics for each branch of the motion plan tree for a high speed vehicle motion. The resulting motion plan therefore implicitly considered uncertainty in its determination of an safe, rapid path across uneven terrain.

Our major results are as follows:

- A method for mobile robot mobility prediction based on the stochastic response surface method was developed. Results obtained using the SRSM matched closely with those obtained through the use of the Monte Carlo methods. The following advantages were observed:
 - Computational time for the SRSM was found to be approximately two orders of magnitude lower than that of classical Monte Carlo methods.
 - The SRSM approach allows for the explicit consideration of uncertainty (in vehicle and/or terrain models) during aggressive maneuvering.
 - The approach represents a potential pathway towards robust aggressive control framework.
- The SRSM method was applied to several practical application scenarios, including mobility prediction on sloped terrain, and high speed motion planning. Major results included:
 - Simulation results of an analysis of vehicle mobility on a side slope.
 - Initial results for a motion planning method that explicitly considers model uncertainty.

Appendix B contains a conference paper that is a direct result of this research.

4.2 Optimal motion planning on heterogeneous terrain.

We consider the problem of computing optimal paths for UGVs from a given configuration to a final configuration. For simplicity, we assumed that the environment is obstacle-free and a simple kinematic model of the car depicted in Figure 3, which is given by the equations

$$\begin{aligned}\dot{x} &= v \cos \theta, \\ \dot{y} &= v \sin \theta, \\ \dot{\theta} &= u,\end{aligned}\tag{1}$$

where $(x, y, \theta) \in SE(2)$, (x, y) is the vehicle's position, θ is the angle between the vehicle and the vertical axis determining the vehicle's orientation, v is the forward, positive velocity and u is the bounded angular acceleration input, which, without loss of generality, is assumed to take value in $[-1, 1]$. This vehicle model is usually referred to as Dubins vehicle and has been broadly used as a kinematic model for path planning of UGVs (and UAVs), like the one depicted in Figure 3.

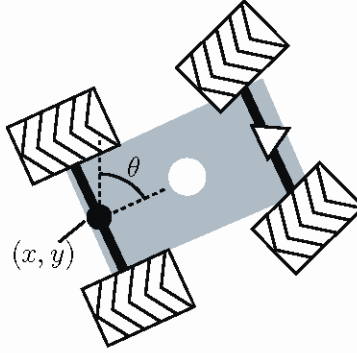


Figure 3: Dubins car.

For a UGV given by (1) on an open terrain, our goal is to compute the optimal path starting at a given initial configuration and ending at a given final configuration. The parameters in the model (1) vary for terrains with different properties. The maximum forward speed changes with the terrain roughness while the maximum curvature of the paths depends on the friction coefficient of the terrain. A convenient modeling abstraction for

navigation on heterogeneous open terrain is to classify terrain regions based on these parameters and then associate to each region a vehicle model with appropriate maximum forward velocity and friction coefficient. For clarity in the exposition, we only consider the case of heterogeneous velocity along the terrain where the vehicle is deployed. Two different velocities, v_1 and v_2 , define the constant, forward velocity of Dubins vehicle on two patches of the plane, patch \mathcal{P}_1 and patch \mathcal{P}_2 , depicted in Figure 4. We are interested in solving the following problem:

Find the minimum-time path for Dubins vehicle from an initial configuration in patch \mathcal{P}_1 to a final configuration in patch \mathcal{P}_2 .

Figure 4 shows possible initial and final vehicle configurations, which are denoted by (x^0, y^0, θ^0) and (x^1, y^1, θ^1) , respectively, for which a minimum-time path is to be found. To the best of our knowledge, the problem described above has not been addressed in the past, perhaps due to the fact that the classical Pontryagin's Maximum Principle is not applicable because of the discontinuous behavior at the common boundary between the patches.

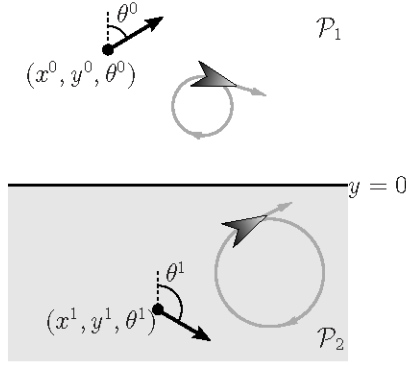


Figure 4: Dubins vehicle on an heterogeneous terrain. The initial configuration is given by (x^0, y^0, θ^0) and the final configuration by (x^1, y^1, θ^1) . The forward velocity in patch \mathcal{P}_1 is smaller than the forward velocity in patch \mathcal{P}_2 .

We have established conditions for time-optimal maneuvers of autonomous vehicles operating on terrain with variable characteristics. Our results include conditions that the paths need to satisfy at the boundary between terrains

with different maximum velocity. These conditions reduce notably the computational load in selecting maneuvers and enable the generation of optimal motion planning algorithms. We summarize them as follows:

- The portions of the paths that remain in either region are Dubins optimal.
- Optimal paths are such that, at the boundary between the regions, their type does not change.
- Optimal paths that cross the boundary describing a straight line are orthogonal to the boundary.
- The angles of the path pieces within each region before and after crossing the boundary satisfy a refraction law.

Appendix B contains formal statements of these results.

4.3 Robust control algorithms for motion primitive selection.

Given a mathematical model of a vehicle, or in more general terms, a dynamical system, with state x and input u (e.g. a static input/output map, an ordinary differential equation with inputs, a partial differential equation with inputs, etc.) describing its behavior for each given input, the motion planning problem for is that of given a *motion plan* v , find a control input u such that the state x satisfies the specification given by the motion plan v . to r . This specification is usually given in terms of a curve in the state space parameterized by time t , in which case the goal of a motion planning algorithm is to steer x to that curve. Algorithms that accomplish such a task are commonly applied in robotics as a method to solve steering problems. One challenge for these algorithms is that in real-world settings, they must accomplish the motion planning task in the presence of obstacles, measurement error, exogenous disturbances, and unmodeled dynamics. Hence, reactivity and robustness are highly desired properties for these algorithms.

A particular class of motion planning algorithms, which are known as *maneuver-based motion planning algorithms*, exploits the symmetry properties present in certain classes of nonlinear systems, in particular, UGVs, to perform challenging motions. *Motion primitives* available in a pre-defined library, designed off-line with model-based design tools, trial and error, and/or

obtained from motions generated by humans, are concatenated to perform a given motion plan. For example, an UGV can be controlled to perform motions, like the one in Figure 5, by breaking the motion in “pieces” where inputs are constants, referred to as *trim trajectories*, and where the inputs are varying with certain law, referred to as *maneuvers*. Figure 5 denotes the different pieces. Typically in maneuver-based motion planners, trim trajectories and maneuvers are executed by applying appropriate open-loop control laws. A method to synchronize the control decisions is to implement a switching logic in a maneuver automaton.

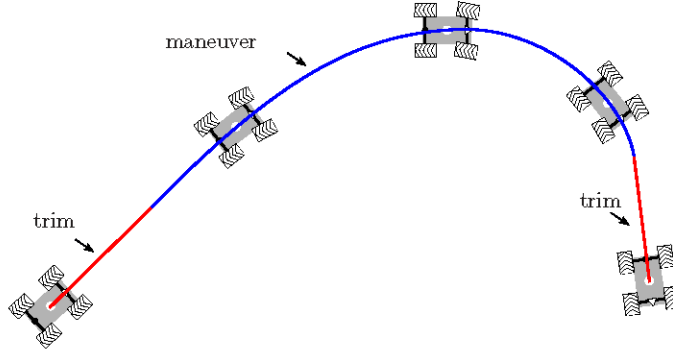


Figure 5: Motion plan for an UGV divided into trim and maneuver pieces.

The open-loop nature of the maneuver-based motion planning method outline above limits its application to nominal scenarios, that is, those without perturbations, e.g. measurement noise, unmodeled dynamics, etc., which, in turn, narrows its applicability to steering of vehicles across open terrain. Even for smooth feedback systems, the presence of arbitrarily small perturbations can lead to totally different behavior than in the nominal case. In fact, establishing robustness (vaguely, the property that under arbitrarily small perturbations the system solutions are “close” to the nominal ones) is not a straightforward task, even in scenarios with homogeneous terrain. Moreover, the metric used to characterize closeness between solutions trajectories (for example, the distance between a trim reference trajectory and a UGV trajectory) should take into account that corresponding pieces in the nominal and perturbed motions may occur over different time intervals.

Another challenge in robustifying maneuver-based motion planning is that the nominal trajectories resulting from it are not always necessarily smooth. Hence, standard trajectory tracking control design techniques are

not applicable. Even when these standard tools are applicable, tracking design for aggressive maneuvers is not always an easy task (for example, to design a tracking controller for an UGV to track motion plans having tight turn maneuvers would require a control law that guarantees fast tracking at the turns while at other pieces of the reference trajectory, slower tracking would suffice).

To address some of the issues mentioned above, we propose a hybrid control framework for robust maneuver-based motion planning. The major results are as follows:

- *Trim trajectories and maneuvers can be combined in a hybrid controller to execute maneuver-based motion plans by means of a timer, two logic variables, and one auxiliary state.*
- *The graphical distance between trajectories specified by the motion planner and the trajectories to the closed-loop (hybrid) system is an appropriate metric to evaluate closeness between motions.*
- *Robustness of tracking of maneuver-based motion planning for general nonlinear systems with symmetries to perturbations in the initial conditions, external disturbances, and unmodeled dynamics.*

We show that this framework results in a hybrid system with implementable semantics, and hence, useful experimental setups. The resulting transition system and control methodology is such that, given a sequence of commands, selects and executes a particular maneuver from a library of motion primitives. The main feature of the system is its robustness to external perturbations, which are typical in controlling UGVs in challenging terrain and, unless the associated control algorithm is robust, concatenation of motion primitives would not be successful. More details can be found in Appendix C.

Regarding the generation of motion primitives, which is key in the construction of a library of trim trajectories and maneuvers for the purposes of motion planning, we provide a detailed calculation of trim trajectories for a particular UGV model: the half car. We show that its set of trim trajectories includes trajectories at the limit of the vehicle’s performance, like those associated to vehicle motions under slipping and skidding conditions occurring at large velocity and small turning radius, which are typical in open terrain settings. See Appendix D for more details.

A Appendix: Mobility Prediction for Unmanned Ground Vehicles in Uncertain Environments

Mobility Prediction for Unmanned Ground Vehicles In Uncertain Environments

Gaurav Kewlani, Karl Iagnemma

Robotic Mobility Group, Massachusetts Institute of Technology, Cambridge, MA, USA

ABSTRACT

The ability of autonomous unmanned ground vehicles (UGVs) to rapidly and effectively predict terrain negotiability is a critical requirement for their use on challenging terrain. Most methods for assessing traversability, however, assume precise knowledge of vehicle and terrain properties. In practical applications, uncertainties are associated with the estimation of the vehicle/terrain parameters, and these uncertainties must be considered while determining vehicular mobility. Here a computationally inexpensive method for efficient mobility prediction based on the stochastic response surface (SRS) approach is presented that considers imprecise knowledge of terrain and vehicle parameters while analyzing various metrics associated with UGV mobility. A conventional Monte Carlo method and the proposed response surface methodology have been applied to two simulated cases of mobility analysis, and it has been shown that the SRS method is an efficient tool as compared to conventional Monte Carlo methods for the analysis of vehicular mobility in uncertain environments.

Keywords: Homogeneous Chaos, Monte Carlo, Latin Hypercube Sampling, Vehicle Mobility, Vehicle Rollover, Stochastic Response Surface, Terrain Modeling, Unmanned Ground Vehicle, Wheel-Soil Interaction

1. INTRODUCTION

Future Army operations will employ autonomous or semi-autonomous unmanned ground vehicles (UGVs) in both cross-country and urban environments. A fundamental requirement of these vehicles is to quickly predict their ability to negotiate rough terrain regions and surmount obstacles. This mobility prediction capability is critical to the successful deployment of UGVs that can operate effectively on challenging terrain with minimal human supervision.

Significant work has been done to understand and predict the mobility of vehicles in natural terrain [1], [2]. However, these efforts assume accurate knowledge of vehicle parameters and wheel- (or track-) soil interaction properties, gathered from terrain measurement devices such as cone penetrometers. In field conditions, however, UGVs often only have access to sparse and uncertain parameter estimates drawn from “standard” robotic sensors such as LIDAR. Moreover, significant uncertainties are often associated with estimates of vehicle parameters, due to effects such as loading, wear, fuel consumption, etc. It is thus critical to consider these uncertainties when deriving predictions of vehicle mobility.

There exists a vast body of literature on techniques to estimate the probability distributions of processes that are subject to uncertainty. Such techniques could be applied to the mobility prediction problem, by first modeling the uncertainty in vehicle and terrain parameters, then defining a range for their probable values, and finally analyzing the performance of a UGV model over that parameter space, as in [3]. The result would be a prediction of the ability of a UGV to successfully traverse a given route that rigorously considers vehicle and terrain parameter uncertainty. This analysis can be performed using a variety of techniques such as interval mathematics, probabilistic methods and fuzzy set theory, among others [4], [5].

A traditional method for estimating the probability density function of a system’s output response from known or estimated input distributions is the Monte Carlo method [6], [7]. This approach involves the random selection of a value for each uncertain parameter from its uncertainty range, weighted by its probability of occurrence, followed by model simulation using this parameter set. This process is repeated many times to obtain the probability distribution of an output metric.

*{gkewlani, kdi}@mit.edu; Phone 617 253-2334; Fax 617 258-7881; <http://web.mit.edu/mobility>

Since parameter values are selected randomly, a large number of simulation runs is often required to obtain reasonable results, leading to a (usually) high computational cost. Structured sampling techniques such as Latin hypercube sampling, importance sampling, and others can be used to improve computational efficiency, however these gains may be modest for complex problems [8], [9].

More recent approaches to stochastic simulation include the polynomial chaos approach, which is based on Wiener's theory of homogeneous chaos. Since the introduction of the spectral stochastic finite element method [10], polynomial chaos has been successfully applied to represent uncertainty in various structural and fluid mechanics problems. Recently, researchers have applied this technique to the dynamic simulation of a 7 DOF vehicle [11]. However, the collocation approach employed therein has been noted to be inherently unstable and exhibit convergence problems [12]. Moreover, different combinations of collocation points may lead to considerably different output estimates, or they may not correspond to high probability regions of the input parameter space.

Here we propose the use of the stochastic response surface method (SRSM), as described by Isukapalli [13], [14] for the mobility prediction of UGVs in natural terrain that uses a regression based approach to obtain an equivalent reduced model for the output and serves as a computationally inexpensive tool for predicting the traversability of a UGV over rugged terrain.

This paper is organized as follows. In Section 2, we briefly introduce the Monte Carlo and SRSM methods and present their application to vehicle dynamic modeling. This is followed by a description of a three degree of freedom vehicle model in Section 3. The effect of terrain physical parameter uncertainty on vehicle mobility is analyzed. Simulation results obtained using Monte Carlo and SRSM approaches are compared in Section 4. It can be seen that accurate, efficient statistical mobility prediction can be achieved using the proposed response surface techniques.

2. UNCERTAINTY ANALYSIS TECHNIQUES

2.1 Monte Carlo Method

With the advancements in computational technology, Monte Carlo techniques have found increasing application in numerous fields over the last several years. These methods typically involve a (usually) large number of simulation runs of an analytical or numerical system model using various combinations of model parameters, followed by the subsequent analysis of the outputs. In other words, the model parameters (known as "input parameters") are randomly sampled from their respective probability distributions, which are assumed to be known (or estimated) a priori, and multiple simulation runs are conducted using each set of the input parameter values to obtain the corresponding outputs for each case. An estimate of the probability distribution of a user-defined output metric can then be estimated.

A variety of methods have been developed for efficient sampling from input parameter probability distributions, including (among others) stratified, importance and Latin Hypercube sampling [15], [16]. Generally, these methods focus on ensuring that samples are generated from the entire range of the input parameter space while reducing computational costs, and are thus an improvement over the standard Monte Carlo method.

In the mobility prediction scenario, vehicle and terrain parameters are designated as uncertain input parameters. A fundamental assumption of the proposed approach is that while the terrain and/or vehicle parameters may not be precisely known, engineering estimates of their distributions are available. This is a reasonable assumption for UGV physical parameter estimates, since the effects of loading, component wear, and parameter uncertainty can generally be bounded with reasonable accuracy. It is also a reasonable assumption for terrain parameter estimates, since many methods exist for coarsely classifying terrain from standard robotic sensors such as LIDAR and vision [17]-[19].

2.1.1 Algorithmic implementation

Here we discuss the general Monte Carlo approach as applied to mobility analysis. The method considers functions of the form:

$$\mathbf{Y} = \mathbf{g}(\mathbf{X}) \quad (1)$$

where \mathbf{g} represents the model under consideration, \mathbf{X} is a vector of uncertain input variables and \mathbf{Y} represents a vector of estimated outputs.

A general procedure for the analysis is as follows:

- a) Construct a vector \mathbf{X} consisting of n relevant terrain and/or vehicle parameters. To define the input parameter space and to characterize the uncertainty in the elements of \mathbf{X} , assign a probability distribution to each input parameter, based on corresponding engineering estimates.

While many forms of the input parameter distribution are possible, in this paper, the parameter values are assumed to have a Gaussian distribution and to be uncorrelated.

- b) Generate a sample value for each of the n input variables from the corresponding probability distribution. More specifically, a sample set:

$$\mathbf{X}_j = [x_{j1}, x_{j2}, \dots, x_{jn}] \quad (2)$$

is generated from the input parameter space. This set may be generated randomly or using the structured sampling techniques such as stratified sampling, importance sampling or Latin Hypercube sampling.

In the “standard” Monte Carlo approach, random sampling of the input parameter distributions is performed. However, to ensure representation of the entire parameter range, a large number of simulations must often be performed. Stratified sampling, on the other hand, partitions the sample space into a number of strata, with each stratum having a specified probability of occurrence. Random samples are then drawn from each stratum. While this ensures dense coverage of the parameter space, the definition of the strata and the calculation of their probabilities must be carefully addressed. Latin hypercube sampling can ensure dense coverage of the range of each input variable while avoiding the difficulties associated with stratified sampling. This is achieved by dividing each input parameter’s range into disjoint intervals of equal probability and then randomly sampling a parameter value from each interval. This is illustrated in Figure 1.

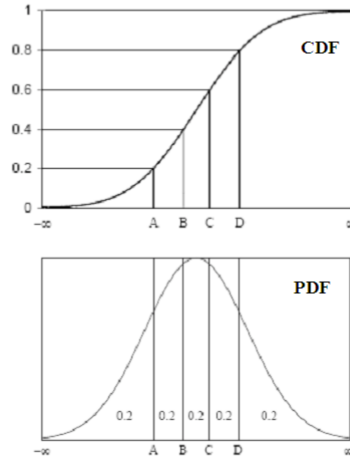


Fig. 1. Illustration of sampling using the Latin hypercube method.

- c) Evaluate the output response from the system model under analysis using the values from the input parameter set \mathbf{X}_j as model parameter values.

- d) Repeat steps b) and c) to generate a probability distribution for the output metric. Various statistics such as the estimated expectation, μ , or variance, σ^2 , can then be determined as follows:

$$\mu = \frac{1}{N} \sum_{j=1}^N g(\mathbf{X}_j) \quad (3)$$

$$\sigma^2 = \frac{1}{N} \sum_{j=1}^N (g(\mathbf{X}_j) - \mu)^2 \quad (4)$$

The number of simulations (N) is chosen to be large enough such that the output distribution converges to a stable value.

Figure 2 below represents schematically the general Monte Carlo approach for uncertainty analysis:

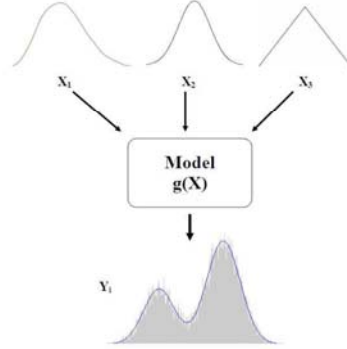


Fig. 2. Illustration of uncertainty analysis using the Monte Carlo method.

2.2 Stochastic Response Surface Method

The stochastic response surface method (SRS) represents inputs and outputs of a system under consideration via series approximations using standard random variables, which results in a computationally efficient means for uncertainty propagation through the models. In SRS, inputs are represented as functions of normal random variables, each having zero mean and unit variance. The same set of random variables that is used to represent input stochasticity can then be used for representation of outputs.

An equivalent reduced model for an output is expressed in the form of a series expansion consisting of multi-dimensional Hermite polynomials of normal random variables, as:

$$y = a_0 + \sum_{i_1=1}^n a_{i_1} \Gamma_1(\xi_{i_1}) + \sum_{i_1=1}^n \sum_{i_2=1}^n a_{i_1 i_2} \Gamma_2(\xi_{i_1}, \xi_{i_2}) + \dots \quad (5)$$

where y refers to an output metric, $a_{i_1}, a_{i_1 i_2}, \dots$ are coefficients to be determined, $\xi_{i_1}, \xi_{i_2}, \dots$ are i.i.d. normal random variables, and $\Gamma_q(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_q})$ is the Hermite polynomial of degree q , given as:

$$\Gamma_q(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_q}) = (-1)^q e^{\frac{1}{2} \mathbf{\xi}^T \mathbf{\xi}} \cdot \frac{\partial^q}{\partial \xi_{i_1} \partial \xi_{i_2} \dots \partial \xi_{i_q}} e^{-\frac{1}{2} \mathbf{\xi}^T \mathbf{\xi}} \quad (6)$$

For notational simplicity, the series may be written as:

$$y(t, \xi) = \sum_{j=0}^N y_j(t) \Phi_j(\xi) \quad (7)$$

where the series is truncated to a finite number of terms and there exists a correspondence between $\Gamma_q(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_q})$ and $\Phi(\xi)$, and their corresponding coefficients.

The series expansion contains unknown coefficient values that can be estimated from a limited number of model simulations to generate an approximate reduced model. This is achieved by choosing a set of sample points from high probability regions then calculating the model output at these points [13], [20]. A regression based approach is then utilized to obtain values for the unknown coefficients [14]. Once the (statistically equivalent) reduced model is formulated, it can be used to facilitate analysis of the system subject to uncertainty.

This procedure thus results in a reduction in the number of model simulations (and, therefore, a reduction in computational cost) required for estimation of output uncertainty, as compared to the conventional probabilistic methods such as Monte Carlo methods. Further details on SRS can be found in [14].

2.2.1 Algorithmic implementation

Here a summary of the SRS method is presented as applied to robotic mobility prediction.

a) Represent uncertain input parameters in terms of standard random variables (here Gaussian variables). A terrain/vehicle parameter X_j can be written as:

$$X_j = \mu_j + \sigma_j \xi \quad (8)$$

where μ_j is the mean, σ_j represents the standard deviation and ξ is a standard normal random variable.

b) Express the model output under consideration in terms of the same set of random variables. While for Gaussian variables, Hermite polynomials are used, different orthogonal polynomial basis functions are used corresponding to the probability distributions of other non-Gaussian variables. This is shown in Table 1.

Table 1. Polynomial Basis Functions and Corresponding Random Variables

RANDOM VARIABLE	POLYNOMIAL FUNCTION
Gaussian	Hermite
Gamma	Laguerre
Beta	Jacobi
Uniform	Legendre

c) Estimate the unknown coefficients of the approximating series expansion. This is accomplished via a regression based approach, first by computing the model output at a set of collocation points [13], [20]. These points are selected such that each standard random variable takes a value of either zero or a root of the Hermite polynomial of a higher order. This ensures that points from high probability regions are represented. Taking the number of collocation points (M) to be nearly twice in number to the number of coefficients ($N+1$) has been shown to yield robust coefficient estimates [14], [20]. Calculation of the model output at these points results in set of equations with the number of equations exceeding the number of unknown coefficients. Using the linear least square method and singular value decomposition, the system of linear equations similar to the one shown below can be solved:

$$\begin{pmatrix} \Gamma_0(\xi_0) & \Gamma_1(\xi_0) & \dots & \Gamma_N(\xi_0) \\ \Gamma_0(\xi_1) & \Gamma_1(\xi_1) & \dots & \Gamma_N(\xi_1) \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_0(\xi_M) & \Gamma_1(\xi_M) & \dots & \Gamma_N(\xi_M) \end{pmatrix} \begin{pmatrix} y_0(t) \\ y_1(t) \\ \vdots \\ y_N(t) \end{pmatrix} = \begin{pmatrix} y(t, \xi_0) \\ y(t, \xi_1) \\ \vdots \\ y(t, \xi_M) \end{pmatrix} \quad (9)$$

The reduced equivalent model can henceforth be used for the analysis, which avoids the requirement of multiple runs of the (generally) non-linear model, thus resulting in reduced simulation time. The advantage of the SRSN technique is therefore that the number of model simulations is greatly reduced relative to conventional methods, thus improving computational efficiency. Further, the accuracy of the computational model can often be increased by increasing the order of the polynomial chaos expansion.

3. 3. MOBILITY PREDICTION SCENARIO

3.1 Traversal over Uncertain Terrain

Here an analysis of a simplified mobile robot terrain traversal scenario is presented using the SRSN technique, which considers a mobile robot traveling on flat outdoor terrain (here modeled as heavy clay) and then attempting to navigate up an inclined region of highly deformable terrain (here modeled as dry sand). This is illustrated in Figure 3. It is assumed that significant uncertainty is associated with a small number of critical terrain physical parameters (here, cohesion and internal friction angle). The UGV's mobility is analyzed using a baseline "standard" Monte Carlo approach (SMC), a Latin hypercube Monte Carlo approach (LHSMC) and the stochastic response surface technique.

A simple description of mobility in the proposed scenario is defined as the probability that for a given initial velocity (\mathbf{u}_0) at the initial position (**A**) (see Figure 3), the robot will have a non-negative velocity at point (**B**), after moving up the sandy incline. Taking into account the uncertainty in terrain parameters, this can be presented as a distribution of traversal probability versus initial velocity, which can then be used to predict for which velocities the robot will be able to traverse the deformable terrain region with a reasonably high probability.

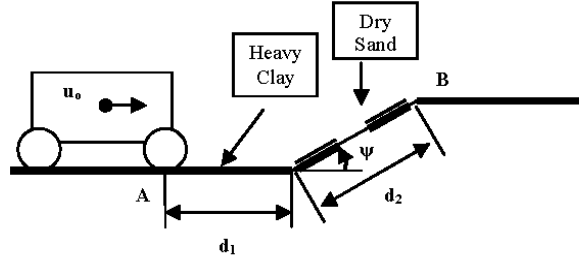


Fig. 3. Simplified scenario considered for mobility prediction under uncertainty

A classical Bekker-type wheel soil interaction model is used to calculate the drawbar pull (i.e. net longitudinal wheel thrust) for the above analysis [21], [22]. This model assumes quasi-static motion, and that the robot wheel is rigid relative to the terrain. An equivalent model for the drawbar pull is then formulated using the approach, which is later used to calculate the traversal probabilities.

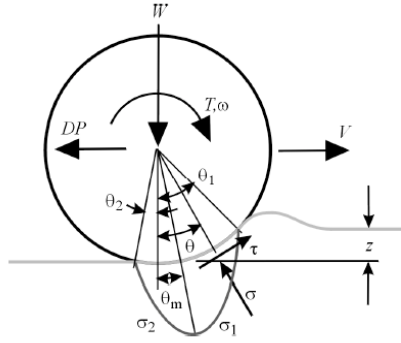


Fig. 4. Wheel-terrain interaction model for rigid wheel on deformable terrain.

For the vehicle terrain interaction model shown in Figure 4, the drawbar pull is given by:

$$DP = rb \left[\int_{\theta_1}^{\theta_2} \tau(\theta) \cos \theta d\theta - \int_{\theta_1}^{\theta_2} \sigma(\theta) \sin \theta d\theta \right] \quad (10)$$

where $\tau(\theta)$ and $\sigma(\theta)$ represent, respectively, the shear stress and normal stress at the wheel-terrain interface (divided into two regions in Figure 4 to more clearly represent the stress distribution), and are given by:

$$\tau(\theta) = (c + \sigma(\theta) \tan \phi) \left(1 - e^{-\frac{r}{k} [1 - \theta - (1-i)(\sin \theta_1 - \sin \theta)]} \right) \quad (11)$$

$$\sigma_1(\theta) = \left(\frac{k_c}{b} + k_\phi \right) (r(\cos \theta - \cos \theta_1))^n \quad (12)$$

$$\sigma_2(\theta) = \left(\frac{k_c}{b} + k_\phi \right) \left[r \left(\cos \left(\theta_1 - \theta \frac{(\theta_1 - \theta_m)}{\theta_m} \right) - \cos \theta_1 \right) \right]^n \quad (13)$$

The drawbar pull can hence be written as:

$$DP = rb \left(\int_0^{\theta_m} \tau_2(\theta) \cos \theta d\theta + \int_{\theta_m}^{\theta_1} \tau_1(\theta) \cos \theta d\theta - \int_0^{\theta_m} \sigma_2(\theta) \sin \theta d\theta - \int_{\theta_m}^{\theta_1} \sigma_1(\theta) \sin \theta d\theta \right) \quad (14)$$

The parameters employed in (10)-(14) are given in Table 2.

Table 2. Parameters Involved In Drawbar Pull Calculation

SYMBOL	QUANTITY
r	Wheel radius
b	Wheel width
θ_1	Angle corresponding to start of contact
θ_2	Angle corresponding to loss of contact
θ_m	Maximum stress angle
c	Cohesion
ϕ	Internal friction angle
i	Wheel slip
n	Sinkage exponent
k_s, k_ϕ	Pressure sinkage moduli

Governing equations of motion for the mobility prediction scenario can now be written as:

$$\dot{x} \, d\dot{x} = \ddot{x} \, dx \quad (15)$$

and

$$\ddot{x} = \frac{DP}{m} - g \sin \psi \quad (16)$$

where m is the vehicle mass, g represents the acceleration due to gravity and ψ is the angle of the incline w.r.t. the horizontal.

3.1.1 Application of SRSIM

As part of this approach, a reduced stochastic model is developed for drawbar pull (DP) considering c and ϕ as uncertain parameters with normal distributions. Cohesion and internal friction angle parameters are represented as:

$$c = \mu_c + \xi_c \sigma_c \quad (17)$$

$$\phi = \mu_\phi + \xi_\phi \sigma_\phi \quad (18)$$

where ξ_c and ξ_ϕ are standard normal random variables. Drawbar pull is now expressed as:

$$DP = a_0 + a_1 \xi_c + a_2 \xi_\phi + a_3 (\xi_c^2 - 1) + a_4 (\xi_\phi^2 - 1) + a_5 \xi_c \xi_\phi \quad (19)$$

The parameters c and ϕ were chosen since they exhibit significant influence on DP . Although they are assumed to be normally distributed, other possible probability distributions (such as uniform or beta distribution) can be considered as well. The corresponding values for c and ϕ used in this analysis can be found in Table 3.

Table 3. Probability Distribution Information For Uncertain Terrain Parameters (c, ϕ)

PARAMETER	DISTRIBUTION FUNCTION	MEAN	STD. DEV.
c (Heavy Clay)	Gaussian	69 kPa	8.50 kPa
ϕ (Heavy Clay)	Gaussian	34 deg	2.10 deg
c (Dry Sand)	Gaussian	1.04 kPa	0.125 kPa
ϕ (Dry Sand)	Gaussian	28 deg	1.75 deg

3.2 Rollover Analysis considering Vehicle Parameter Uncertainty

Here a three degree of freedom vehicle model (see Figure 5) is considered that includes lateral acceleration, yaw and roll dynamics. The roll and yaw moments of inertia are represented by I_{xx} and I_{zz} respectively, m is the total vehicle mass, m_s is the sprung mass and v is the longitudinal velocity of the vehicle. The front wheel steering angle is represented by δ . The linearized equations for this model are given as:

$$mv(\dot{\beta} + \dot{\psi}) - m_s h \ddot{\varphi} = \sum F_y = C_f(\delta - \frac{\dot{\psi} l_f}{v} - \beta) + C_r(\frac{\dot{\psi} l_r}{v} - \beta) \quad (21)$$

$$I_{zz} \dot{\psi} = \sum M_z = C_f(\delta - \frac{\dot{\psi} l_f}{v} - \beta) l_f - C_r(\frac{\dot{\psi} l_r}{v} - \beta) l_r \quad (22)$$

$$(I_{xx} + m_s h^2) \ddot{\varphi} = \sum M_x = m_s g h \varphi + m_s h v (\dot{\beta} + \dot{\psi}) + M_s \quad (23)$$

where C_f and C_r are the cornering stiffnesses of the lumped front and rear wheels, and l_f and l_r are respectively the distances of the front and rear axles from the cg. The suspension moment, represented by M_s , is given as:

$$M_s = -(k_f + k_r) \varphi - (b_f + b_r) \dot{\varphi} \quad (24)$$

where k_f and k_r are the stiffnesses and b_f and b_r are the damping factors for the front and rear axles respectively. It should be noted that there is uncertainty associated with the estimation of the values of the above vehicle parameters. This uncertainty will be considered in the present analysis.

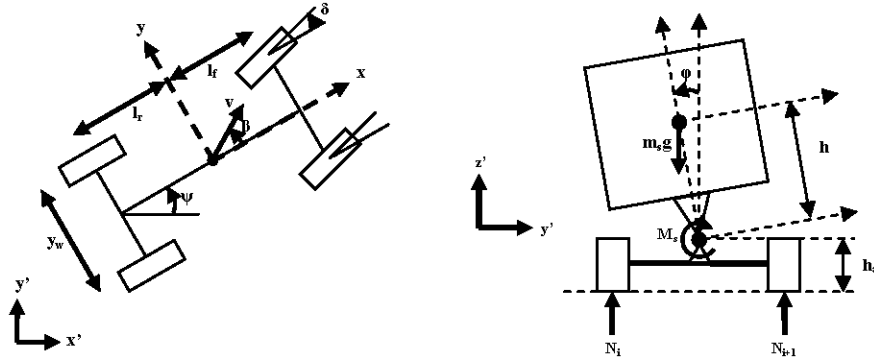


Fig. 5. Vehicle Model for Mobility Analysis

For measuring vehicle mobility, a simple rollover coefficient is employed that is similar to the one described in [24]. Using the principle of balance of moments and vertical forces, a rollover metric for the linear model above is given by:

$$R = \frac{2m_s}{mg y_w} (h_a + h) \left(v(\dot{\beta} + \dot{\psi}) - h \ddot{\varphi} \right) \quad (25)$$

where h_a is the height of the roll axis above the ground and y_w is the track width. For this metric, absolute values of R that are greater than 1 indicate vehicle wheel liftoff and thus impending rollover.

3.2.1 Application of SRSM

We define the state space, \mathbf{X} as:

$$\mathbf{X} = \begin{bmatrix} \beta & \dot{\psi} & \varphi & \dot{\varphi} \end{bmatrix}^T \quad (26)$$

The variables related to suspension stiffness are represented as polynomial chaos expansions, using Hermite polynomials of standard normal random variables. Here, ξ_1 and ξ_2 , which are used to represent the input uncertainty in the system. The front and rear axle roll stiffness are considered to be normally distributed about their mean values. This is represented as:

$$k_f = \mu_{k_f} + \xi_1 \sigma_{k_f} \quad (27)$$

$$k_r = \mu_{k_r} + \xi_2 \sigma_{k_r} \quad (28)$$

Then the state variables can be represented as:

$$\beta(t, \xi) = \sum_{j=0}^P \beta_j(t) \Phi_j(\xi) \quad (29)$$

$$\dot{\psi}(t, \xi) = \sum_{j=0}^P \dot{\psi}_j(t) \Phi_j(\xi) \quad (30)$$

$$\varphi(t, \xi) = \sum_{j=0}^P \varphi_j(t) \Phi_j(\xi) \quad (31)$$

$$\dot{\varphi}(t, \xi) = \sum_{j=0}^P \dot{\varphi}_j(t) \Phi_j(\xi) \quad (32)$$

where $\xi = [\xi_1, \xi_2]$.

The parameter values considered for the steering stiffness are shown in Table 4.

Table 4. Uncertain Vehicle Parameters in Rollover Analysis

PARAMETER	MEAN (Nm/rad)	STD. DEV. (Nm/rad)
k_f	30×10^3	4×10^3
k_r	30×10^3	4×10^3

A spectral stochastic analysis [11], [20] is performed using the above expansions to obtain the time evolution of the rollover coefficient, subject to various steering input functions (sinusoidal, ramp-like and a double lane change maneuver).

4. RESULTS

4.1 Traversal over Uncertain Terrain

Results from analysis of the mobility prediction scenario described in Section 3.1 are presented using SMC, LHSMC and SRSM methods are presented for inclination angles (ψ) of 6° and 15° (see Figure 6). It can be seen that the distributions generated by the SRSM method are nearly identical to those generated by the SMC and LHSMC methods. Increasing the number of SMC and LHSMC runs slightly decreases distribution variance, however in both cases the differences among the three methods is small.

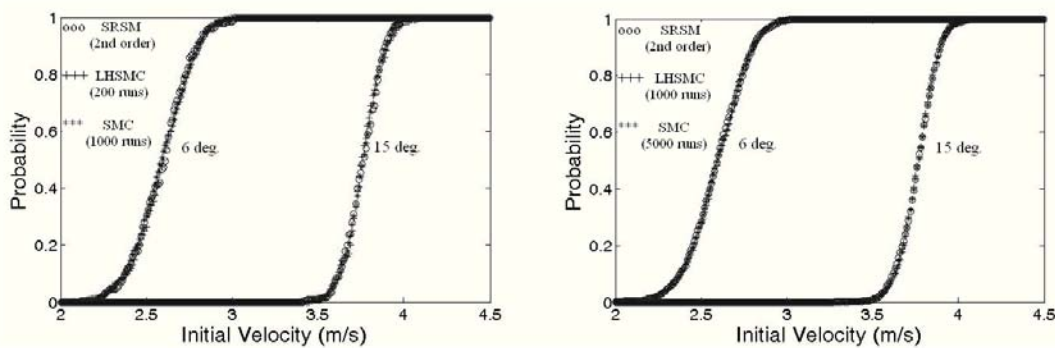


Fig. 6. Probability plots for mobility prediction scenario, for small number of SMC and LHSMC runs (left) and large number (right). The coefficient values obtained for a 2nd order expansion of drawbar pull (19) considering c , φ as the uncertain parameters were:

$$a_0 = -1.4260, a_1 = 0.2981, a_2 = 0.5586, a_3 = 0.0000, a_4 = 0.0091, a_5 = 0.0000.$$

The results predict that increasing the robot's initial velocity increases the probability of safe slope traversal, as expected. Also, the minimum initial velocity required for successful traversal increases as the inclination increases. A clearly defined "transition region" can be observed, where the probability of safe traversal is a function of terrain parameter variance as well. This region effectively describes the "risk" of traversal at a certain critical velocity range.

Also, the computation time of the SRSM method is compared to SMC and LHSMC in Table 5, for the case of a terrain inclination angle of 15 degrees. It can be seen that the proposed approach results in a significant computational reduction compared to the baseline approaches. This suggests that on-line, real time implementation of the method is feasible for simple models.

Table 5. Computation Time for Mobility Prediction Analysis

METHOD	SIMULATION RUNS	TIME TAKEN (sec)
SMC	5000	55.750
LHSMC	1000	11.343
SRSM (2 nd order)		0.406

4.2 Rollover Analysis considering Vehicle Parameter Uncertainty

Results from analysis of the rollover scenario described in Section 3.2 are presented here. Simulations for various vehicle maneuvers were conducted using the stochastic response surface method (SRSM), standard Monte Carlo (SMC) and Monte Carlo Latin Hypercube Sampling (LHSMC). The accuracy of the SRSM is compared to that of the SMC and LHSMC and the plots are presented in Figures 7 - 9. Close agreement between the three methods can be observed.

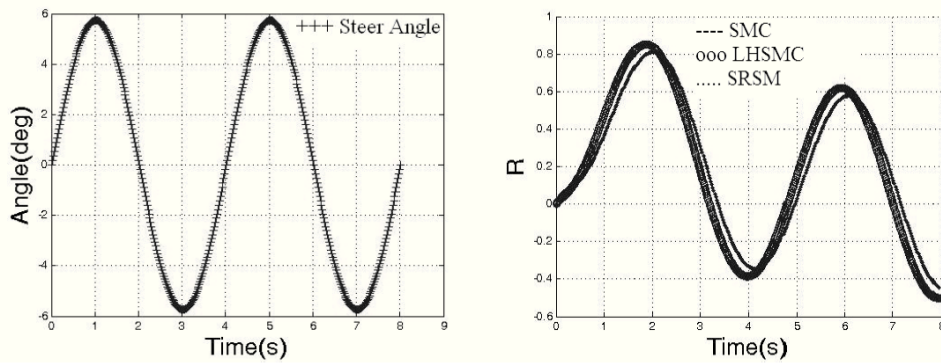


Fig. 7. Vehicle Rollover Analysis for Sinusoidal Steering Input using Various Statistical Analysis Techniques

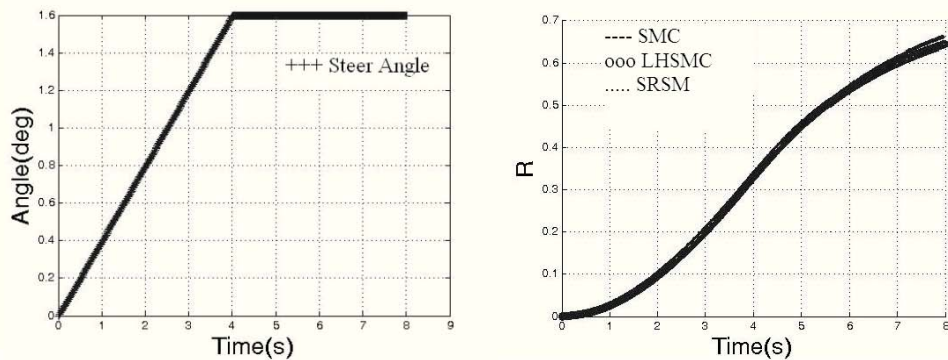


Fig. 8. Vehicle Rollover Analysis for Ramp-Like Steering Input using Various Statistical Analysis Techniques

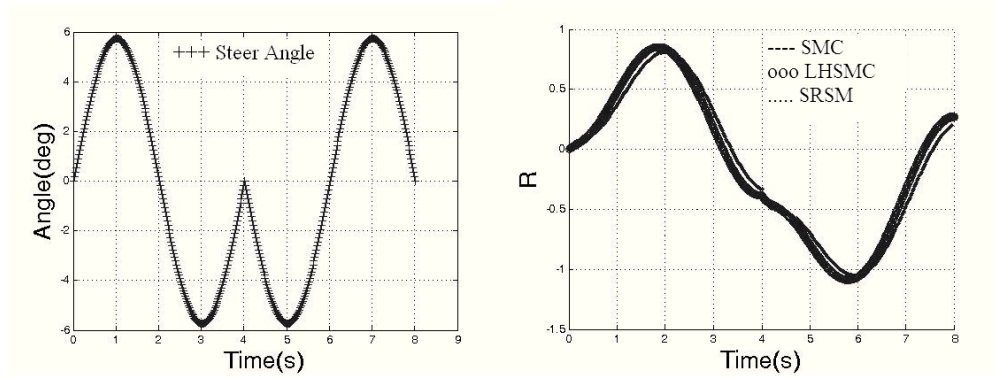


Fig. 9. Vehicle Rollover Analysis for Double Lane Change Steering Input using Various Statistical Analysis Techniques

Stochastic analysis allows insight into the range of the variation of an output time series. In Figure 10, results are shown for the steering angle and rollover coefficient for a double lane change maneuver, here including uncertainty bounds on the 2σ variation. In this particular analysis, it can be observed that while the mean absolute value of the rollover metric (corresponding closely to the result from a deterministic simulation) remains less than one, the uncertain value exceeds one, thus indicating a substantial risk of vehicle rollover when parameter uncertainty is explicitly considered.

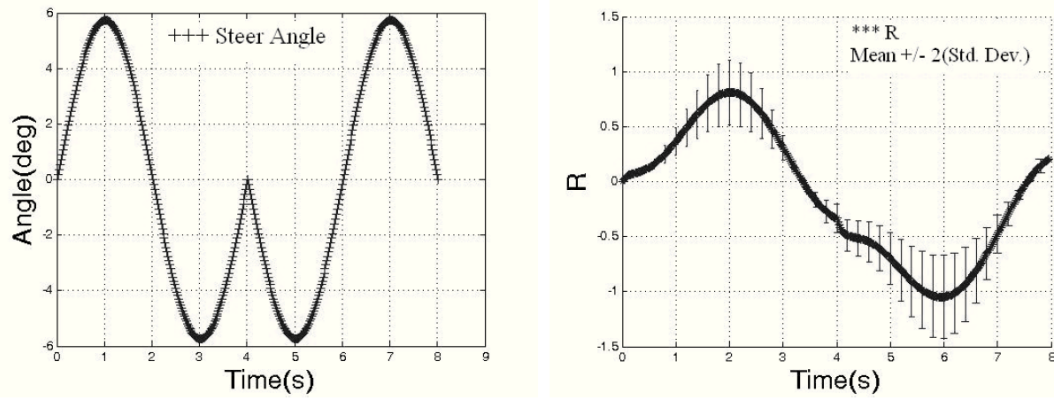


Fig. 10 Vehicle Rollover Analysis Using SRSM

The simulation times for the approaches are also compared (see Table 6) for the various steering inputs. Computation time for the SRSM method is approximately two orders of magnitude lower than for LHSMC. All computations were performed on a desktop PC running unoptimized Matlab code.

Table 6. Simulation Times using Various Statistical Analysis Techniques for Vehicle Rollover Analysis

STEERING INPUT	SMC (2000 RUNS)	LHSMC (400 RUNS)	SRSM
Sinusoidal	1920.1 s	793.688 s	6.691 s
Ramp-like	1933.5 s	796.219 s	6.766 s
Double Lane Change	1952.7 s	808.235 s	6.797 s

4. 5. CONCLUSION

This paper has presented an approach to statistical mobile robot mobility prediction based on the stochastic response surface method. This approach explicitly considers uncertainty present in vehicle and terrain physical parameter estimates. Simulation results of simplified mobility prediction scenarios have shown that the proposed method represents a significant improvement over conventional Monte Carlo methods in terms of computational efficiency, and

can be used for robustly and efficiently predicting the traversability of mobile robots in unstructured environments. Current work is focused on statistical modeling of more complex three dimensional UGV models.

REFERENCES

- [1] Jurkat, M., Nuttall, C., and Haley, P., *The AMC '74 Mobility Model*, Technical Report 11921, US Army Tank Automotive Command, Warren, MI, 1975
- [2] Ahlvin, R.B. and Haley, P.W., *NATO Reference Mobility Model Edition II, NRM User's Guide*, Technical Report GL-92-19, U.S. Army WES, Vicksburg, MS, 1992.
- [3] Golda, D., Iagnemma, K., and Dubowsky, S., "Probabilistic Modeling and Analysis of High-Speed Rough-Terrain Mobile Robots," *Proceedings of the IEEE International Conference on Robotics and Automation*, 2004.
- [4] Helton, J.C., "Uncertainty and Sensitivity Analysis Techniques for Use in Performance Assessment for Radioactive Waste Disposal", *Reliability Engineering and System Safety*, 42, 327-367, 1993.
- [5] Hanss, M., "The transformation method for the simulation and analysis of systems with uncertain parameters", *Fuzzy Sets and Systems*, v.130 n.3, p.277-289.
- [6] Rubinstein, R. Y., *Simulation and the Monte Carlo Method*, John Wiley & Sons, Inc., New York, NY, 1981.
- [7] Kalos, M. H., Whitlock, P. A., *Monte Carlo methods*. Vol. 1: Basics, Wiley-Interscience, New York, NY, 1986.
- [8] McKay, M. D., "Latin hypercube sampling as a tool in uncertainty analysis of computer models", *Proceedings of the 24th conference on Winter simulation*, p.557-564, 1992, Arlington, Virginia, United States.
- [9] Helton, J. C., Davis, F. J., "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems", *Reliability Engineering and System Safety* 81 (1), 23-69, 2003
- [10] Ghanem, R., Spanos, P. D., *Stochastic Finite Elements: A Spectral Approach*, Springer-Verlag, New York, 1991.
- [11] Li, L., Sandu, C., "On the impact of cargo weight, vehicle parameters, and terrain characteristics on the prediction of traction for off-road vehicles", *Journal of Terramechanics*, Volume 44, Issue 3, July 2007, p.221-238.
- [12] Atkinson, K. E., *An Introduction to Numerical Analysis*, Second Edition, John Wiley & Sons, New York, 1988.
- [13] Isukapalli, S.S., Balakrishnan, S., Georgopoulos, P.G., "Computationally efficient uncertainty propagation and reduction using the stochastic response surface method", *IEEE Conference on Decision and Control*, p. 2237-2243, 2004.
- [14] Isukapalli, S. S., *Uncertainty analysis of transport-transformation models*, Ph.D. Thesis, Rutgers University, 1999.
- [15] Helton, J. C., Johnson, J. D., Sallaberry, C. J., Storlie, C. B., "Survey of sampling-based methods for uncertainty and sensitivity analysis", *Reliability Engineering and Systems Safety*, 91, 1175-1209, 2006.
- [16] McKay, M. D., Beckman, R. J., Conover, W. J., "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code", *Technometrics*, 21, 239-245, 1979.
- [17] Vandapel, N., Huber, D.F., Kapuria, A., Hebert, M. (2004). "Natural Terrain Classification using 3-D Ladar Data", *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 5, 5117- 5122.
- [18] Manduchi, R., Castano, A., Thalukder, A., and Matthies, L. (2005, May). "Obstacle detection and terrain classification for autonomous off-road navigation," *Autonomous Robots*, 18, 81-102.
- [19] Halatci, I., Brooks, C., and Iagnemma, K., "Terrain Classification and Classifier Fusion for Planetary Exploration Rovers," *Proceedings of the IEEE Aerospace Conference*, 2007.
- [20] Huang, S., Mahadevan, S., Rebba, R., "Collocation-based stochastic finite element analysis for random field problems", *Probabilistic Engineering Mechanics*, 22, 194-205, 2007.
- [21] Bekker, G., *Theory of Land Locomotion*, University of Michigan Press, 1956.
- [22] Bekker, G., *Introduction to Terrain-Vehicle Systems*, University of Michigan Press, 1969.
- [23] Iagnemma, K., Shibly, H., Dubowsky, S., "On-Line Terrain Parameter Estimation for Planetary Rovers", *IEEE International Conference on Robotics and Automation*, 2002.
- [24] Odenthal, D., Bunte, T., Ackermann, J., "Nonlinear steering and braking control for vehicle rollover avoidance", *European Control Conference*, 1999, Karlsruhe, Germany.

B Appendix:
 On the Optimality of Dubins Paths across
 Heterogeneous Terrain

On the Optimality of Dubins Paths across Heterogeneous Terrain

Ricardo G. Sanfelice and Emilio Frazzoli

Abstract

We derive optimality conditions for the paths of a Dubins vehicle when the state space is partitioned into two patches with different vehicle's forward velocity. We recast this problem as a hybrid optimal control problem and solve it using optimality principles for hybrid systems. Among the optimality conditions, we derive a “refraction” law at the boundary of the patches which generalizes the so-called Snell's law of refraction in optics to the case of paths with bounded maximum curvature.

1 Introduction

Control algorithms that are capable of steering autonomous vehicles to satisfy a given set of specifications, like initial and final constraints, and at the same time, guarantee certain optimality conditions are very appealing to applications in robotics and aerospace. This has led researchers to strive for control design tools that adequately incorporate both trajectory constraints and measures of optimality. As a consequence, many results from the theory of optimal control, in particular, those that guarantee time optimality, have found wide applicability in autonomous vehicle control problems.

Perhaps, the earliest result on time-optimal control laws for autonomous vehicles modeled as a particle moving with constant, positive forward velocity and with constrained minimum turning radius is the work by Dubins [6]. While Dubins used only geometric arguments to establish his results, a few years later, the appearance of Pontryagin's Maximum Principle in [11] enabled the authors in [3] to systematically recover Dubins results. Moreover,

building from the work of Reeds and Shepp [12], the application of Pontryagin's optimality principle permitted the authors in [20, 3] to derive similar results for a vehicle model without forward velocity constraints.

In this paper, we consider autonomous vehicles with dynamics governed by

$$|u| \leq 1, \quad \begin{cases} \dot{x} &= v \sin \theta \\ \dot{y} &= v \cos \theta \\ \dot{\theta} &= u \end{cases}, \quad (1)$$

where (x, y) is the vehicle's position, θ is the angle between the vehicle and the vertical axis determining the vehicle's orientation, u is the angular acceleration input for the vehicle, and v is the vehicle's forward velocity. This vehicle model is usually referred to as Dubins vehicle. We consider the case of heterogeneous velocity along the terrain where the vehicle is deployed. Two different velocities, v_1 and v_2 , define the constant, forward velocity of Dubins vehicle on two patches of the plane, patch \mathcal{P}_1 and patch \mathcal{P}_2 , depicted in Figure 1. We are interested in the following problem:

Find the minimum-time path for Dubins vehicle from an initial point and angle in patch \mathcal{P}_1 to a final point and angle in patch \mathcal{P}_2 .

Figure 1 shows possible initial and final vehicle configurations, which are denoted by (x^0, y^0, θ^0) and (x^1, y^1, θ^1) , respectively, for which a minimum-time path is to be found. To the best of our knowledge, the problem described above has not been addressed in the past, perhaps due to the fact that the classical Pontryagin's Maximum Principle is not applicable because of the discontinuous behavior at the common boundary between the patches.

By recasting this problem into an optimal hybrid control problem and applying principles of optimality for hybrid systems, we establish the following conditions that illuminate important characteristics of optimal paths:

- *The portions of the paths that remain in either patch are Dubins optimal.*
- *Optimal paths are such that, at the boundary between the patches, their type does not change; that is, the type of path right before and after crossing the boundary are the same.*

- *Optimal paths that cross the boundary describing a straight line are orthogonal to the boundary.*
- *The angles of the path pieces before and after crossing the boundary satisfy a “refraction” law, which consists of a generalization of Snell’s law of refraction in optics.*

Applications of these results include optimal motion planning of autonomous vehicles in environments with obstacles, different terrains properties, and other topological constraints. Strategies that steer autonomous vehicles across heterogeneous terrain using Snell’s law of refraction have already been recognized in the literature and applied to point-mass vehicles; see, e.g., [1, 13]. Our results extend those to the case of autonomous vehicles with Dubins dynamics.

The remainder of the paper is organized as follows. Section 2 discusses related background to the optimal control problem outlined above and introduces general notation. In Section 3, we present a hybrid model which, as shown in that same section, enable us to formulate the problem of study in an optimal hybrid control framework. In Section 4, we establish necessary conditions for optimality of paths including a refraction law at the boundary of the patches. Due to space constraints, the technical proofs are omitted and will be published elsewhere.

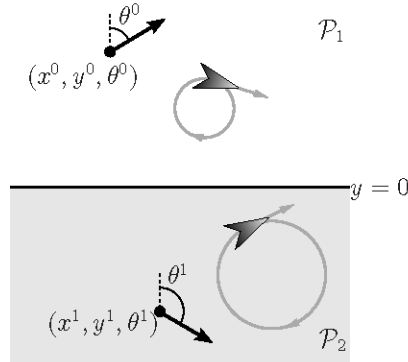


Figure 1: Dubins vehicle on an heterogeneous terrain. The initial configuration is given by (x^0, y^0, θ^0) and the final configuration by (x^1, y^1, θ^1) . The forward velocity in patch \mathcal{P}_1 is smaller than the forward velocity in patch \mathcal{P}_2 .

2 Background

Pontryagin's Maximum Principle [11] is a very powerful tool to derive necessary conditions for optimality of solutions to a dynamical system. In words, this principle establishes the existence of an adjoint function with the property that, along optimal system solutions, the Hamiltonian obtained by combining the system dynamics and the cost function associated to the optimal control problem is minimized. In its original form, this principle is applicable to optimal control problems with dynamics governed by differential equations with continuously differentiable right-hand sides.

The shortest path problem between two points with specific tangent direction and bounded maximum curvature has received wide attention in the literature. In his pioneer work in [6], by means of geometric arguments, Dubins showed that optimal paths to this problem consist of a smooth concatenation of no more than three pieces, each of them describing either a straight line, denoted by \mathcal{L} , or a circle, denoted by \mathcal{C} (when the circle is traveled clockwise, we write \mathcal{C}^+ , while when the circle is traveled counter-clockwise, we write \mathcal{C}^-), and are either of type \mathcal{CCC} or \mathcal{CLC} , that is, they are among the following six types of paths

$$\mathcal{C}^-\mathcal{C}^+\mathcal{C}^-, \mathcal{C}^+\mathcal{C}^-\mathcal{C}^+, \mathcal{C}^-\mathcal{LC}^-, \mathcal{C}^+\mathcal{LC}^+, \mathcal{C}^+\mathcal{LC}^-, \mathcal{C}^-\mathcal{LC}^+, \quad (2)$$

in addition to any of the subpaths obtained when some of the pieces (but not all) have zero length. More recently, the authors in [3] recovered Dubins' result by using Pontryagin's Maximum Principle; see also [20]. Further investigations of the properties of optimal paths to this problem and other related applications of Pontryagin's Maximum Principle include [16, 2, 4], to just list a few.

Optimal control problems exhibiting discontinuous/impulsive behavior, like the heterogeneous version of Dubins' problem outlined in Section 1, cannot be solved using the classical Pontryagin's Maximum Principle. Extensions of this principle to systems with discontinuous right-hand side appeared in [17] while extensions to hybrid systems include [18], [7], and [15]. These principles establish the existence of an adjoint function which, in addition to conditions that parallel the necessary optimality conditions in the principle by Pontryagin, satisfies certain conditions at times of discontinuous/jumping behavior. The applicability of these principles to relevant problems have been highlighted in [18, 10, 5]. These will be the key tool in deriving the results in this paper.

2.1 Notation

We use the following notation throughout the paper. \mathbb{R}^n denotes n -dimensional Euclidean space. \mathbb{R} denotes the real numbers. $\mathbb{R}_{\geq 0}$ denotes the nonnegative real numbers, i.e., $\mathbb{R}_{\geq 0} = [0, \infty)$. \mathbb{N} denotes the natural numbers including 0, i.e., $\mathbb{N} = \{0, 1, \dots\}$. Given $k \in \mathbb{N}$, $\mathbb{N}_{\leq k}$ denotes $\{0, 1, \dots, k\}$. Given a set S , \overline{S} denotes its closure and S° denotes its interior. Given a vector $x \in \mathbb{R}^n$, $|x|$ denotes the Euclidean vector norm. Given $U := [-1, 1]$, \mathcal{U} denotes the set of all piecewise-continuous functions u from subsets of $\mathbb{R}_{\geq 0}$ to U .

3 Problem Statement

In this section, we formulate the problem of steering Dubins vehicle across heterogeneous terrain as a hybrid optimal control problem. We present a hybrid model and introduce the optimal control problem. An alternative approach is to treat this problem as a differential equation with discontinuous right-hand side and use the results in [17]. However, a hybrid control systems approach is not only more convenient from a modeling point of view as it enables the use of a sound concept of solution but also facilitates the application of more explicit optimality principles for hybrid systems, like the ones in [18].

3.1 Hybrid model

We denote by \mathcal{H}_v the hybrid system that captures the dynamics of Dubins vehicle along the patches. Let $v_1, v_2 \in \mathbb{R}_{>0}$, $v_1 \neq v_2$, be the forward velocity of the vehicle on patch \mathcal{P}_1 and patch \mathcal{P}_2 , respectively, where

$$\mathcal{P}_1 := \{[x \ y \ \theta]^\top \in \mathbb{R}^3 \mid y \geq 0\} \ , \quad \mathcal{P}_2 := \{[x \ y \ \theta]^\top \in \mathbb{R}^3 \mid y \leq 0\} \ ,$$

which share a common boundary $\mathcal{P}_1 \cap \mathcal{P}_2 = \{[x \ y \ \theta]^\top \in \mathbb{R}^3 \mid y = 0\}$; see Figure 1. Let q be a discrete state taking value in $Q := \{1, 2\}$ that indicates the current patch to which the vehicle belongs to. Following the vehicle's dynamics in (1),

$$\begin{bmatrix} \dot{\xi} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} f_q(\xi, u) \\ 0 \end{bmatrix} \quad \xi \in \mathcal{P}_q \tag{3}$$

with

$$\xi := \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \in \mathbb{R}^3 \quad \text{and} \quad f_q(\xi, u) := \begin{bmatrix} v_q \sin \theta \\ v_q \cos \theta \\ u \end{bmatrix}$$

define the continuous dynamics (or *flows*) of \mathcal{H}_v , where ξ is the continuous state and $u \in \mathcal{U}$ is the control input. Then, during flows, ξ captures the vehicle dynamics on the q -th patch while q remains constant. We model the change of patch so that it occurs when y is zero and the vehicle is moving away from the current patch. Then, defining a function $s : Q \rightarrow \{-1, 1\}$ where $s(1) = -1$ and $s(2) = 1$, the discrete dynamics (or *jumps*) of \mathcal{H}_v are given by

$$\begin{bmatrix} \xi^+ \\ q^+ \end{bmatrix} = \begin{bmatrix} \xi \\ 3 - q \end{bmatrix} \quad \xi \in \mathcal{P}_1 \cap \mathcal{P}_2 \quad \text{and} \quad s(q)v_q \cos \theta > 0, \quad (4)$$

which implies that at jumps ξ does not change while q is toggled between 1 and 2. Finally, we denote by $\zeta := [\xi^\top q]^\top$ the full state of \mathcal{H}_v .

Following the hybrid systems framework outlined in [8] and further established in [9, 14], we can rewrite \mathcal{H}_v as

$$\mathcal{H}_v : \quad \begin{cases} \dot{\zeta} &= f(\zeta, u) & \zeta \in C \\ \zeta^+ &= g(\zeta) & \zeta \in D \end{cases}$$

by defining

$$f(\zeta, u) := \begin{bmatrix} f_q(\xi, u) \\ 0 \end{bmatrix}, C := \bigcup_{q \in Q} (C_q \times \{q\}),$$

$$g(\zeta) := \begin{bmatrix} \xi \\ 3 - q \end{bmatrix}, D := \bigcup_{q \in Q} (D_q \times \{q\}),$$

where $C_q := \mathcal{P}_q$ and $D_q := \{\xi \in \mathbb{R}^3 \mid y = 0, s(q)v_q \cos \theta > 0\}$ for each $q \in Q$. Then, \mathcal{H}_v is determined by the data (f, C, g, D) , where f is the *flow map*, C is the *flow set*, g is the *jump map*, and D is the *jump set*. As in [8], solutions to \mathcal{H}_v are given by *hybrid arcs* on *hybrid time domains*. Hybrid time domains use a variable t to indicate flow time and an index j to keep track of the number of jumps, and hence, parameterize solutions by (t, j) . A subset E of $\mathbb{R}_{\geq 0} \times \mathbb{N}$ is a *hybrid time domain* if it is the union of infinitely many intervals of

the form $[t_j, t_{j+1}] \times \{j\}$, where $0 = t_0 \leq t_1 \leq t_2 \leq \dots$, or of finitely many such intervals, with the last one possibly of the form $[t_j, t_{j+1}] \times \{j\}$, $[t_j, t_{j+1}) \times \{j\}$, or $[t_j, \infty) \times \{j\}$. (Note that the t component of elements $(t, j) \in E$ does not uniquely define the index j since, in this framework, multiple jumps at the same t are possible.) Then, given a control input $u \in \mathcal{U}$, solutions to \mathcal{H}_v are given by functions, called *hybrid arcs*, $\zeta : \text{dom } \zeta \rightarrow \mathbb{R}^4$, where $\text{dom } \zeta$ is a hybrid time domain, $t \mapsto \xi(t, j)$ is a locally absolutely continuous function for each fixed j , $t \mapsto q(t, j)$ is a piecewise constant function for each fixed j , and ζ satisfies the flow and jump conditions mentioned above. More precisely, given an input $u \in \mathcal{U}$, a hybrid arc ζ is a *solution to the hybrid system* \mathcal{H}_v if $\zeta(0, 0) \in C \cup D$, $\text{dom } \zeta = \text{dom } u$, and:

(S1) For all $j \in \mathbb{N}$ and almost all t such that $(t, j) \in \text{dom } \zeta$ ¹,

$$\zeta(t, j) \in C, \quad \dot{\zeta}(t, j) = f(\zeta(t, j), u(t, j)) .$$

(S2) For all $(t, j) \in \text{dom } \zeta$ such that $(t, j+1) \in \text{dom } \zeta$,

$$\zeta(t, j) \in D, \quad \zeta(t, j+1) = g(\zeta(t, j)) .$$

Inputs u given as signals $t \mapsto u(t)$ for each $t \in \mathbb{R}_{\geq 0}$ can be rewritten on a hybrid time domain E by defining, with some abuse of notation, $u(t, j) := u(t)$ for each $(t, j) \in E$. Note that solutions to \mathcal{H}_v exist from every point in $C \cup D = \mathbb{R}^3 \times Q$. In particular, solutions are allowed to flow in the boundary $\mathcal{P}_1 \cap \mathcal{P}_2$ with either $q = 1$ or $q = 2$; such a feature cannot be captured with a differential equation with discontinuous right-hand side or with a (regular) differential inclusion without adding extra solutions. Also note that since the sets D_q are not closed subsets of \mathbb{R}^3 , the regularity property for D required in [9, 14] does not hold (the flow map, jump map, and jump set of \mathcal{H}_v satisfy the properties therein). While such a regularity is not required for the results in this paper to be true, it turns out that, as shown in [14], it highlights the presence of undesirable solutions if the sets D_q were to be closed or small noise entered through the state.

3.2 Hybrid optimal control problem

We consider the following hybrid optimal control problem. Given $(x^0, y^0, \theta^0) \in C_1^\circ$ and $(x^1, y^1, \theta^1) \in C_2^\circ$:

¹ $\dot{\zeta}(t, j)$ denotes the derivative of $t \mapsto \zeta(t, j)$ with respect to t for a fixed j , which exists for almost every t such that $(t, j) \in \text{dom } \zeta \cap ([t_j, t_{j+1}] \times \{j\})$.

(\star) Minimize the transfer time $T \in \mathbb{R}_{\geq 0}$ subject to:

(C1) Dynamical constraint: dynamics of \mathcal{H}_v given in (3)-(4).

(C2) Input constraint: $u \in \mathcal{U}$.

(C3) Initial and terminal constraints: every optimal solution (ξ, q) to \mathcal{H}_v satisfies the initial constraint $(x(0, 0), y(0, 0), \theta(0, 0)) = (x^0, y^0, \theta^0)$ and the terminal constraint $(x(T, J), y(T, J), \theta(T, J)) = (x^1, y^1, \theta^1)$ for some $(T, J) \in \text{dom}(\xi, q)$.

The number of jumps required to solve (\star) is finite, given by $J - 1$, and no smaller than one; hence, optimal solutions to (\star) are not Zeno. The initial and final constraints are such that solutions can flow from some time before their first jump and after their final jump (that is, the first jump is at some $(t_1, 0)$ with $t_1 > 0$ and the last jump is at some $(t_J, J - 1)$ with $t_J < T$). This is a technical requirement for the application of the hybrid maximum principle in [18] in the next section.

4 Necessary conditions for optimality

Necessary optimality conditions for solutions to \mathcal{H}_v solving (\star) can be obtained using the principle of optimality for hybrid systems in [18] (see also [19] and [10]). Under further technical assumptions, Theorem 1 in [18] establishes that there exists an *adjoint pair* (λ, λ_o) , where λ is a function and λ_o is a constant, which, along optimal solutions to (\star), satisfies certain *Hamiltonian maximization, nontriviality, transversality, and Hamiltonian value conditions*. In particular, [18, Theorem 1] can be applied to the optimal control problem (\star) to deduce the following optimality conditions for the paths.

Proposition 4.1 *[properties of (\star)] For each optimal solution (ξ, q) to (\star) with optimal control u , minimum transfer time T , and $J-1$ number of jumps, there exists a function $\lambda : \text{dom } \lambda \rightarrow \mathbb{R}^3$, $\lambda := [\alpha \ \beta \ \gamma]^\top$, $\text{dom } \lambda = \text{dom}(\xi, q)$, where $t \mapsto \lambda(t, j)$ is absolutely continuous for each j , $(t, j) \in \text{dom } \lambda$, and a constant $\lambda_o \in \mathbb{R}$ defining the adjoint pair (λ, λ_o) satisfying:*

$$a) \ \lambda_o \geq 0 \text{ and } \dot{\lambda}(t, j) = -\frac{\partial H_{q(t, j)}}{\partial \xi}(\xi(t, j), \lambda(t, j), \lambda_o, u(t, j)) \text{ for almost every } t \in [t_j, t_{j+1}], \ (t, j) \in \text{dom } \lambda, \text{ where, for each } q \in \{1, 2\}, \ H_q :$$

$\mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R} \times U \rightarrow \mathbb{R}$ is the Hamiltonian associated with the continuous dynamics of \mathcal{H}_v , which is given by

$$H_q(\xi, \lambda, \lambda_o, u) = \alpha v_q \sin \theta + \beta v_q \cos \theta + \gamma u - \lambda_o$$

for each $q \in Q$.

- b) There exist $\bar{\alpha}, \bar{\beta} \in \mathbb{R}$ and, for each $j \in \mathbb{N}_{\leq J}$, there exists $p_j \in \mathbb{R}$ such that $\alpha(t, j) := \bar{\alpha}$ for all $(t, j) \in \text{dom}(\xi, q)$, $\beta(t, j) := \bar{\beta} + p_j$ for almost all $t \in [0, T]$, $(t, j) \in \text{dom}(\xi, q)$, and $\gamma(t, j) = \gamma(t, j+1)$ for each (t, j) such that $(t, j), (t, j+1) \in \text{dom } \lambda$.
- c) For every $(t, j) \in \text{dom}(\xi, q)$ such that $\gamma(t, j) \neq 0$, $u(t, j) = \text{sgn}(\gamma(t, j))$; and for every $(t, j) \in \text{dom}(\xi, q)$ such that $\gamma(t, j) = 0$, $u(t, j) = 0$.
- d) For every $(t, j) \in \text{dom}(\xi, q)$ such that $\gamma(t, j) = 0$, $\beta(t, j) \tan \theta(t, j) = \alpha(t, j)$.

Remark 4.2 The proof of Proposition 4.1 uses the fact that \mathcal{H}_v can be associated with a hybrid system \mathcal{H}_v^* given in the framework in [18] and that every solution to \mathcal{H}_v solving (\star) is also a solution to \mathcal{H}_v^* (agreeing with the concept of solution in [18]²). This property follows by construction of \mathcal{H}_v^* . Hybrid systems in [18] and [10] have a continuous state ξ with flows governed by $\dot{\xi} = f_q(\xi, u)$ when ξ belongs to a smooth manifold \mathcal{M}_q , where $q \in Q$ is a discrete state (which remains constant during flows). Jumps from mode q to mode q' satisfy: 1) the switching condition $(\xi, \xi') \in S_{q,q'}$, where ξ is the continuous state before the jump, ξ' is the continuous state after the jump, and $S_{q,q'}$ is the switching set; and 2) a temporal constraint enforcing that the

²In [18], solutions to hybrid systems are given on compact time intervals by absolutely continuous functions ξ_j on $[t_j, t_{j+1}]$ such that, for each $j \in \{1, 2, \dots, \nu\}$ (with finite $\nu \in \mathbb{N}$) and for finite sequences of logic states $\{q_j\}$ and control inputs $\{u_j\}$, satisfy the flow condition $\dot{\xi}_j = f_{q_j}(\xi_j(t), u_j(t))$ for almost all $t \in [t_j, t_{j+1}]$ and the jump condition $(\xi_j(t_j), \xi_{j+1}(t_j)) \in S_{q_j, q_{j+1}}$ for each t_j , where t_j denotes the jump time (which is assumed to belong to the interior of the compact time interval where solutions are defined) and $S_{q_j, q_{j+1}}$ is the switching set at the j -th jump (see [18, Definition 3] for more details). Hence, passing from a solution ζ on a bounded hybrid time domain $\text{dom } \zeta$ with jumps at different (t_j, j) 's, first jump at $(t_1, 0)$ with $t_1 > t_0$, and last jump at $(t_J, J-1)$ with $t_J < T$, where $T := \sup \{t \in \mathbb{R}_{\geq 0} \mid \exists j \in \mathbb{N} \text{ such that } (t, j) \in \text{dom } \zeta\}$ and $J := \sup \{j \in \mathbb{N} \mid \exists t \in \mathbb{R}_{\geq 0} \text{ such that } (t, j) \in \text{dom } \zeta\}$, to a solution as in [18, Definition 3] is straightforward.

jump time for the current mode is in the set $J_q \subset \mathbb{R}$. To obtain \mathcal{H}_v^* , the sets C_q in \mathcal{H}_v are replaced by smooth manifolds \mathcal{M}_q , $C_q \subset \mathcal{M}_q$, while the jump set and the jump map are replaced by the switching condition given by

$$\mathcal{S}_{1,2} = \mathcal{S}_{2,1} = \hat{\mathcal{S}} := \{(\xi, \xi) \mid y = 0, \xi \in \mathbb{R}^3\} ,$$

and $J_1 = J_2 = \mathbb{R}$. Then, the properties of the adjoint pair guaranteed by [18, Theorem 1] automatically imply item a) in Proposition 4.1 (see [18, Definition 9]). The condition for optimality at switches for the adjoint state λ implies that only the second component of λ , i.e. β , has a jump while the other two components are continuous (see Remark 4.3). This implies item b) in Proposition 4.1. The Hamiltonian maximization condition guaranteed to hold by [18, Theorem 1] implies that

$$H_{q(t,j)}(\xi(t,j), \lambda(t,j), \lambda_o, u(t,j)) = \max_{w \in U} H_{q(t,j)}(\xi(t,j), \lambda(t,j), \lambda_o, w)$$

for almost every $t \in [t_j, t_{j+1}]$, $(t, j) \in \text{dom } \lambda$ (see [18, Definition 10]). It follows that the control law in item c) in Proposition 4.1 maximizes H_q . By integrating the adjoint state λ when $u = 0$, Proposition 4.1.d follows automatically. ■

Remark 4.3 [18, Theorem 1] implies that at jumps, the optimal solution, optimal control, and adjoint pair satisfy the switching condition $(-\lambda(t, j), \lambda(t, j+1)) \in K_j^\perp$ for each j for which there exists $t \in [0, T]$ such that $(t, j), (t, j+1) \in \text{dom } \lambda$, where K_j^\perp is the polar of the Boltyanskii approximating cone to $\mathcal{S}_{q(t,j), q(t,j+1)} (= \hat{\mathcal{S}})$. The set $\hat{\mathcal{S}}$ is such that K_j^\perp is given by

$$\left\{ w \in \mathbb{R}^3 \times \mathbb{R}^3 \mid \langle w, v \rangle \leq 0 \ \forall v \in \hat{\mathcal{S}} \right\}$$

since the Boltyanskii approximating cone to $\hat{\mathcal{S}}$ is the set itself. Then, since by definition of $\hat{\mathcal{S}}$ the second and fourth components of v in K_j^\perp are zero, $(-\lambda(t, j), \lambda(t, j+1)) \in K_j^\perp$ if and only if $\alpha(t, j) = \alpha(t, j+1)$, $\gamma(t, j) = \gamma(t, j+1)$, which implies that only β can have a jump. This property can also be obtained using the optimality principles in [15]. ■

4.1 Optimality of paths

The properties of the adjoint pair (λ, λ_o) and the control input u in Proposition 4.1 can be related to properties of the continuous component ξ of the solutions to (\star) . These characterize the optimal paths from given initial and terminal constraints, as the following theorem states.

Theorem 4.4 *[optimality conditions of solutions to (\star)] Each optimal solution (ξ, q) to (\star) with optimal control u , minimum transfer time T , and $J-1$ number of jumps is such that:*

- a) *The continuous component ξ is a smooth concatenation of finitely many pieces from the set $\{C^+, C^-, \mathcal{L}\}$.*
- b) *The input component u is piecewise constant with finitely many pieces taking value in $\{-1, 0, 1\}$.*
- c) *Each piece of the continuous component ξ contained in C_q , $q \in Q$, is Dubins optimal between the first and last point of such piece, i.e., it is given as in (2).*
- d) *For each $(t, j) \in \text{dom}(\xi, q)$ for which $(x(t, j), y(t, j), \theta(t, j)) \in D_{q(t, j)}$, the solution has a jump and:*
 - d.1) *If the path before the jump is C then the path after the jump is C .*
 - d.2) *If the path before the jump is \mathcal{L} then the path after the jump is \mathcal{L} and $\theta(t, j)$ is zero or any multiple of π .*

Remark 4.5 *The proof of Theorem 4.4 uses Proposition 4.1 and the fact that, since the jump condition in \mathcal{H}_v is time independent (that is, $J_1 = J_2 = \mathbb{R}$), the Hamiltonian value condition guaranteed to hold by [18, Theorem 1] implies that there exists $h^* \in \mathbb{R}$ such that*

$$h^* = H_{q(t, j)}(\xi(t, j), \lambda(t, j), \lambda_o, u(t, j))$$

for almost every $t \in [t_j, t_{j+1}]$, $(t, j) \in \text{dom } \lambda$ (see [18, Definition 13]). ■

Figure 2 depicts optimal paths around the boundary of the patches. Item d.1) in Theorem 4.4 implies that optimal paths that cross the boundary are of the same type at each side of it. More precisely, if before crossing the boundary, the optimal path is of type \mathcal{C} (\mathcal{C}^+ or \mathcal{C}^-), then the optimal path after crossing the boundary is also of type \mathcal{C} (\mathcal{C}^+ or \mathcal{C}^- , respectively). Figure 2(a) depicts an optimal path of type \mathcal{C}^+ . Statement d.2) in Theorem 4.4 implies that \mathcal{L} -type paths at the boundary are optimal only if they are orthogonal to the boundary. Figure 2(b) depicts this situation.

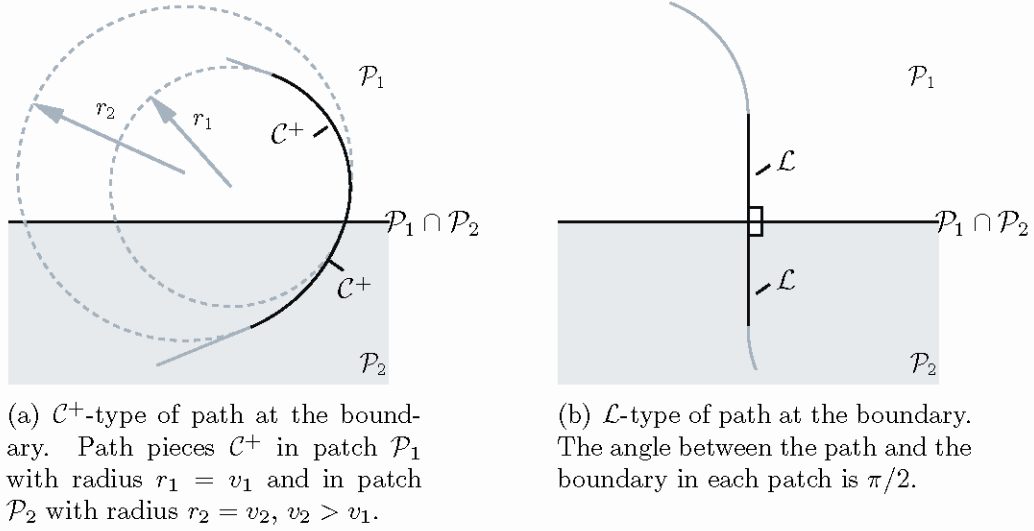


Figure 2: Optimal paths nearby the boundary: paths of types \mathcal{C}^+ and \mathcal{L} satisfying the necessary conditions in Theorem 4.4.

Using Theorem 4.4, it is possible to determine optimal families of paths for a class of solutions to (\star) . The following statements follow directly from Dubins' result and Theorem 4.4.

Corollary 4.6 *[optimal paths w/one jump] Every optimal solution (ξ, q) to (\star) with only one jump is such that the continuous component ξ is a smooth concatenation of \mathcal{C}, \mathcal{L} paths pieces and is given by one of the following four types of paths*

$$\mathcal{C}_1 \mathcal{L}_1 \mathcal{C}_2 \mathcal{L}_2 \mathcal{C}_3, \mathcal{C}_1 \mathcal{C}_2 \mathcal{C}_3 \mathcal{L}_1 \mathcal{C}_4, \mathcal{C}_1 \mathcal{C}_2 \mathcal{C}_3 \mathcal{C}_4 \mathcal{C}_5, \mathcal{C}_1 \mathcal{L}_1 \mathcal{C}_2 \mathcal{C}_3 \mathcal{C}_4, \quad (5)$$

in addition to any such path obtained when some of the path pieces (but not all) have zero length. Furthermore, if the path piece intersecting the boundary is of type \mathcal{L} , then the continuous component ξ describes a path of type $C_1\mathcal{L}_1C_2$ (or any such path obtained when C_1 and/or C_2 have zero length).

A consequence of Theorem 4.4 that is useful when computing optimal paths is the following.

Corollary 4.7 [nonoptimal paths] *For the optimal control problem (\star) , solutions to \mathcal{H}_v satisfying (C1)-(C3) with the continuous component ξ describing paths that change at the boundary are nonoptimal, that is, paths that before and after the boundary are given by C^+ and \mathcal{L} , C^- and \mathcal{L} , \mathcal{L} and C^+ , \mathcal{L} and C^- , C^+ and C^- , or C^- and C^+ , respectively, are nonoptimal.*

Figure 3 depicts two of the path types that Corollary 4.7 determines to be nonoptimal.

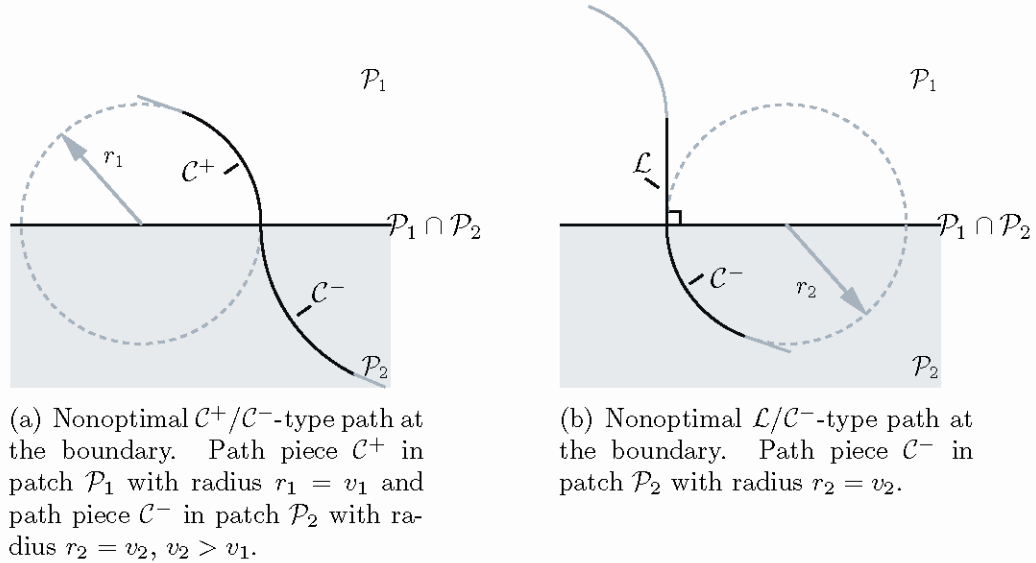


Figure 3: Nonoptimal paths at the boundary: paths of type C^+/C^- and \mathcal{L}/C^- changing at the boundary and hence, not satisfying the necessary conditions for optimality in Theorem 4.4.

4.2 Refraction law at boundary

The optimal control law given in Proposition 4.1.c and the properties of the component γ of the adjoint state λ given in Proposition 4.1.b imply that the control law is constant at jumps of \mathcal{H}_v (note that u is piecewise continuous for each fixed j with discontinuities at (t, j) 's where the path type changes). While θ remains constant at the boundary, the initial and final angles (and their variations) of the paths intersecting the boundary satisfy the following algebraic condition involving the patch velocities v_1 and v_2 .

Theorem 4.8 *[refraction law for (\star)] Let (ξ, q) be an optimal solution to (\star) . Let θ_1 and θ_2 denote the initial and final angle, respectively, of a path piece intersecting the boundary $\mathcal{P}_1 \cap \mathcal{P}_2$, as show in Figure 4. Let $\Delta\theta_1, \Delta\theta_2 \in \mathbb{R}$ be given by $\Delta\theta_1 := \theta^* - \theta_1$, $\Delta\theta_2 := \theta_2 - \theta^*$, where θ^* is the angle between the path and the boundary $\mathcal{P}_1 \cap \mathcal{P}_2$ at their intersection (with respect to the vertical axis). If the path piece intersecting $\mathcal{P}_1 \cap \mathcal{P}_2$ is of type \mathcal{C} , then $v_1, v_2, \theta_1, \theta_2, \Delta\theta_1$ and $\Delta\theta_2$ satisfy*

$$\frac{v_1}{v_2} = \frac{1 + \cot \theta_2 \cot \left(\frac{\Delta\theta_1 - \Delta\theta_2}{2} + \frac{\theta_1 + \theta_2}{2} \right)}{1 + \cot \theta_1 \cot \left(\frac{\Delta\theta_1 - \Delta\theta_2}{2} + \frac{\theta_1 + \theta_2}{2} \right)}, \quad (6)$$

and if the path piece intersecting $\mathcal{P}_1 \cap \mathcal{P}_2$ is of type \mathcal{L} , then θ_1 and θ_2 are equal to π .

Remark 4.9 Equation (6) in Theorem 4.8 implies that for a path of type \mathcal{C} intersecting $\mathcal{P}_1 \cap \mathcal{P}_2$ to be optimal, $\theta_1, \theta_2, \Delta\theta_1$ and $\Delta\theta_2$ shown in Figure 4 must satisfy (6). When the path intersecting $\mathcal{P}_1 \cap \mathcal{P}_2$ is of type \mathcal{L} , by Corollary 4.6, the path \mathcal{L} is orthogonal to $\mathcal{P}_1 \cap \mathcal{P}_2$ and consequently, there is no “refraction” at the boundary. This is depicted in Figure 2(b). The proof of Theorem 4.8 follows from the properties of the optimal solution and adjoint state at jumps stated in Theorem 4.4 and Proposition 4.1.d. \blacksquare

Equation (6) can be interpreted as a refraction law at the boundary of the two patches for the angles (and their variations) θ_1, θ_2 (and $\Delta\theta_1, \Delta\theta_2$). This parallels Snell’s law of refraction in optics, which states a relationship between the angles of rays of light when passing through the boundary of two isotropic media with different refraction coefficients. More precisely, given

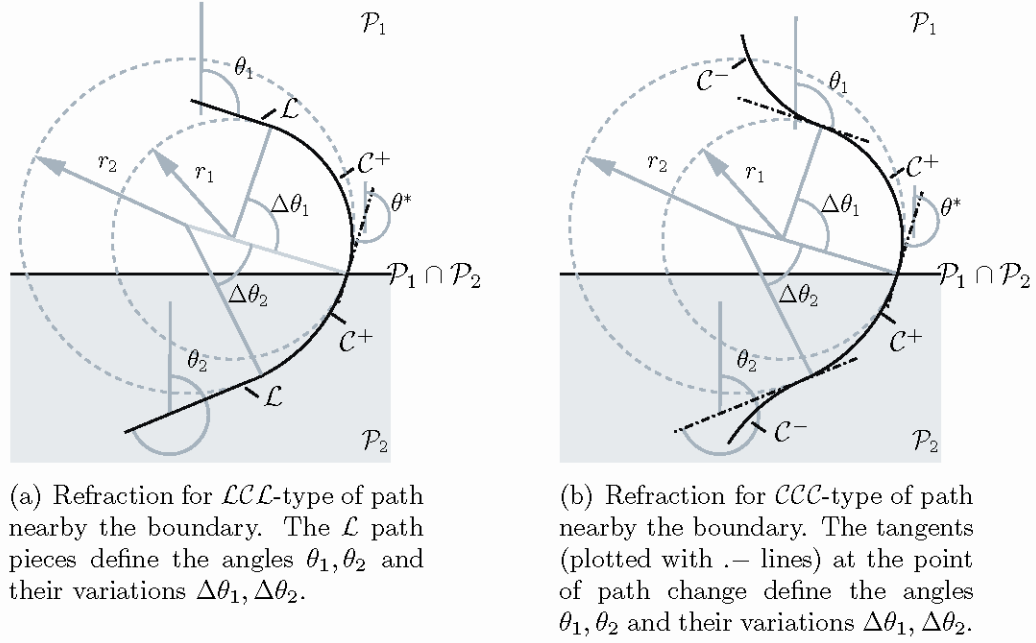


Figure 4: Refraction law for paths at the boundary. The initial and final angles of optimal paths intersecting the boundary given by θ_1 and θ_2 , respectively, and their variations $(\Delta\theta_1, \Delta\theta_2)$ satisfy equation (6), which is a generalization of Snell's law of refraction.

two media with different refraction indexes v_1 and v_2 , Snell's law of refraction states that

$$\frac{v_1}{v_2} = \frac{\sin \theta_1}{\sin \theta_2}, \quad (7)$$

where θ_1 is the angle of incidence and θ_2 is the angle of refraction. This law can be derived by solving a minimum-time problem between two points, one in each medium. Moreover, the dynamics of the rays of light can be associated to the differential equations $\dot{x} = v_i$, where v_i is the velocity in the i -th medium, $i = 1, 2$. Theorem 4.8 generalizes Snell's law to the case when the dynamics of the rays of light are given by (1). In fact, (6) reduces to (7) when $\Delta\theta_1 = \theta_1$ and $\Delta\theta_2 = \theta_2$. In the context of autonomous vehicles, (6) consists of a generalization of the refraction law for optimal steering of a point-mass vehicle, as in [1, 13], to the Dubins vehicle case.

To further illustrate our results, consider $v_1 = 2v_2 > 0$, (x^0, y^0, θ^0) , and (x^1, y^1, θ^1) as depicted in Figure 5. A path corresponding to a solution to \mathcal{H}_v

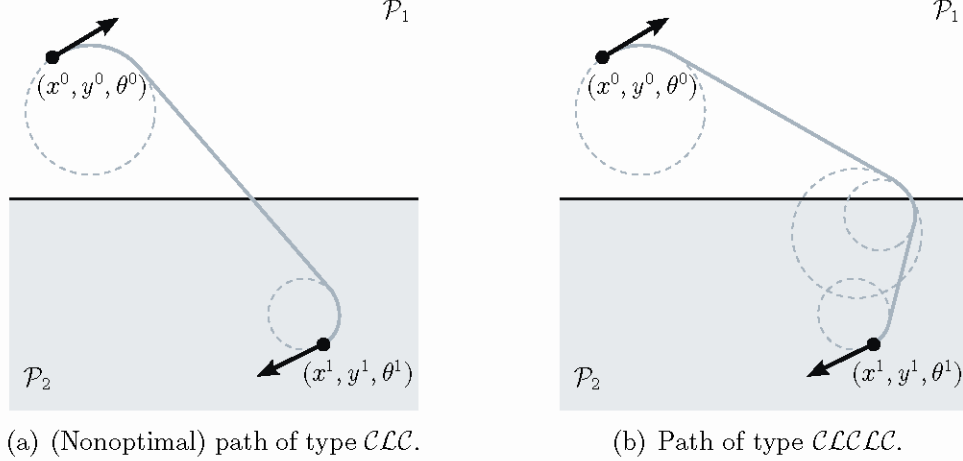


Figure 5: Optimal control of Dubins vehicle on patches with velocities $v_1 = 2v_2$. The path depicted in (a) is nonoptimal since its \mathcal{L} -type piece is not orthogonal to the boundary $\mathcal{P}_1 \cap \mathcal{P}_2$ (it is also nonoptimal since it does not exploit the fact that the maximum velocity in patch \mathcal{P}_1 is twice faster than in patch \mathcal{P}_2). The path depicted in (b) is a candidate for optimality as it satisfies the conditions in Theorem 4.4 and Corollary 4.6.

matching the initial and terminal constraints is shown in Figure 5(a). Since the \mathcal{L} -type path piece smoothly connecting the \mathcal{C} -type paths at (x^0, y^0, θ^0) and (x^1, y^1, θ^1) does not intersect the boundary $\mathcal{P}_1 \cap \mathcal{P}_2$ orthogonally, Theorem 4.4.d implies that it is nonoptimal (see also Corollary 4.6). Note that this path is not taking advantage of the fact that in patch \mathcal{P}_1 , the vehicle can travel twice faster than in patch \mathcal{P}_2 . Paths candidate for being optimal are like the one depicted in Figure 5(b) as it satisfies the conditions in Theorem 4.4 and Corollary 4.6.

5 Conclusions

We have derived necessary conditions for the optimality of paths with bounded maximum curvature. To establish our results, we formulated the problem as

a hybrid optimal control problem and used optimality principles from the literature. Our results provide verifiable conditions for optimality of paths. These include conditions both in the interior of the patches and at their common boundary, as well as a refraction law for the angles which generalizes Snell’s law of refraction in optics to the current setting. Applications of our results include optimal motion planning tasks for autonomous vehicles with Dubins vehicle dynamics.

References

- [1] R. S. Alexander and N. C. Rowe. Path planning by optimal-path-map construction for homogeneous-cost two-dimensional regions. *Proc. IEEE International Conference on Robotics and Automation*, pages 1924–1929, 1990.
- [2] D. J. Balkcom and M. T. Mason. Time optimal trajectories for bounded velocity differential drive vehicles. *The International J. Robotics Research*, 21:199–217, 2002.
- [3] J. D. Boissonnat, A. Cérézo, and J. Leblond. Shortest paths of bounded curvature in the plane. *Journal of Intelligent and Robotic Systems*, 11:5–20, 1994.
- [4] H. Chitsaz, S. M. LaValle, D. J. Balkcom, and M. T. Mason. Minimum wheel-rotation paths for differential-drive mobile robots. *Proceedings IEEE International Conference on Robotics and Automation*, 2006.
- [5] C. D’Apice, M. Garavello, R. Manzo, and B. Piccoli. Hybrid optimal control: case study of a car with gears. *International Journal of Control*, 76:1272–1284, 2003.
- [6] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79:497–516, 1957.
- [7] M. Garavello and B. Piccoli. Hybrid necessary principle. *SIAM J. Control Optim.*, 43(5):1867–1887, 2005.

- [8] R. Goebel, J.P. Hespanha, A.R. Teel, C. Cai, and R.G. Sanfelice. Hybrid systems: generalized solutions and robust stability. In *Proc. 6th IFAC Symposium in Nonlinear Control Systems*, pages 1–12, 2004.
- [9] R. Goebel and A.R. Teel. Solutions to hybrid inclusions via set and graphical convergence with stability theory applications. *Automatica*, 42(4):573–587, 2006.
- [10] B. Piccoli. Necessary conditions for hybrid optimization. In *Proc. 38th IEEE Conference on Decision and Control*, pages 410–415, 1999.
- [11] L. S. Pontryagin, V. G. Boltyanskij, R. V. Gamkrelidze, and E. F. Mishchenko. *The mathematical theory of optimal processes*. Wiley, 1962.
- [12] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145:367–393, 1990.
- [13] N. C. Rowe and R. S. Alexander. Finding optimal-path maps for path planning across weighted regions. *The International Journal of Robotics Research*, 19:83–95, 2000.
- [14] R.G. Sanfelice, R. Goebel, and A.R. Teel. Generalized solutions to hybrid dynamical systems. *To appear in ESAIM: Control, Optimisation and Calculus of Variations*, 2008.
- [15] M. S. Shaikh and P. E. Caines. On the hybrid optimal control problem: Theory and algorithms. *IEEE Transactions on Automatic Control*, 52:1587–1603, 2007.
- [16] A. M. Shkel and V. Lumelsky. Classification of the Dubins’ set. *Robotics and Autonomous Systems*, 34:179–202, 2001.
- [17] H. J. Sussmann. Some recent results on the maximum principle of optimal control theory. *Systems and Control in the Twenty-First Century*, pages 351–372, 1997.
- [18] H. J. Sussmann. A maximum principle for hybrid optimal control problems. In *Proc. 38th IEEE Conference on Decision and Control*, pages 425–430, 1999.

- [19] H. J. Sussmann. *Stability and Stabilization of Nonlinear Systems, Lecture Notes in Control and Information Sciences*, chapter A nonsmooth hybrid maximum principle, pages 325–354. Springer, 1999.
- [20] H. J. Sussmann and G. Tang. Shortest paths for the Reeds-Shepp car a worked out example of the use of geometric techniques in nonlinear optimal control. Technical report, Rutgers Center for Systems and Control Technical Report, 1991.

C Appendix:
 A Hybrid Control Framework for Robust
 Maneuver-Based Motion Planning

A Hybrid Control Framework for Robust Maneuver-based Motion Planning

Ricardo G. Sanfelice and Emilio Frazzoli

Abstract

We introduce a modeling framework for robustness of maneuver-based motion planning algorithms for nonlinear systems with symmetries. Our framework implements a hybrid controller that robustly combines motion primitives, which consist of trim trajectories and maneuvers, from a pre-defined library. The closed-loop system is viewed as a hybrid system with flows given by a differential equation, jumps given by a difference equation, and two sets where these dynamics are allowed. We show that our hybrid controller for implementation of motion planning algorithms confers to the closed-loop system robustness properties to a large class of perturbations.

1 Introduction

Motion planning algorithms are commonly applied in robotics as a method to solve steering problems. In a real-world scenario, the motion planning task needs to be accomplished in the presence of obstacles, measurement error, exogenous disturbances, and unmodeled dynamics. To guarantee some degree of robustness, motion planning algorithms are usually blended with feedback control algorithms, which track the output of the motion planner; see, e.g., [1, 11, 12, 14, 18].

The motion planning problem itself is typically recast as an optimal control problem with cost function and constraints stemming from the given task to be accomplished along with its specifications. In complex motion planning problems, online computation of optimal control policies is not always feasible. A motion planning technique suitable in such cases was proposed in [6]

for general nonlinear systems with symmetries. A motion plan in [6] is given by a concatenation of a finite number of *motion primitives* selected from a pre-defined library and implemented in a *maneuver automaton*. Motion primitives were defined in [6] as equivalence classes of trajectories, induced by symmetries in the system's dynamics, e.g., invariance with respect to time, translations, and rotations.

One of the main features of the maneuver-motion based approach is that each element in the motion primitives library can be designed off-line subject to particular specifications, like optimality, state constraints, etc., relaxing in this way on-line computation requirements; see, e.g., its applications to robotics in [5, 7, 18]. However, this method combines motion primitives in an open-loop manner, which restricts its application to nominal scenarios, that is, those without perturbations. Moreover, the fact that the trajectories resulting from this approach are not necessarily smooth, renders the task of robustifying motion plans via feedback control challenging since standard trajectory tracking control design techniques are not applicable.

In this paper, we propose a hybrid control algorithm that executes maneuver-based motion plans and combines state feedback control laws for nonlinear systems with symmetries. The purpose of our hybrid controller is to provide a control framework for maneuver-based motion planning featuring robustness properties to perturbations. We show that this framework results in a hybrid system with implementable semantics, and hence, useful experimental setups. This class of hybrid systems has been recently introduced in [8, 9] motivated by the pursue of robustness of asymptotic stability. Our control framework for maneuver-based motion planning also borrows ideas from the techniques in [16] for robust combination of state feedback and open-loop controllers, and also from the invariant constructions in [4].

The paper is organized as follows. Section 2 introduces notation and basic definitions regarding nonlinear systems with symmetries, motion primitives and plans, and hybrid systems. Section 3 introduces our hybrid control framework for motion planning, while Section 4 states its main properties.

2 Preliminaries

2.1 Notation

\mathbb{R} denotes the real numbers. $\mathbb{R}_{\geq 0}$ denotes the nonnegative real numbers, i.e., $\mathbb{R}_{\geq 0} = [0, \infty)$. \mathbb{N} denotes the natural numbers including 0, i.e., $\mathbb{N} = \{0, 1, \dots\}$. $\mathbb{N}_{< k}$ ($\mathbb{N}_{\leq k}$) denotes numbers in \mathbb{N} from 0 to $k - 1$ (from 0 to k , respectively). \mathbb{R}^n denotes the n -dimensional Euclidean space. \mathbb{B} denotes the open unit ball in a Euclidean space. Given a set S , \overline{S} denotes its closure and S° denotes its interior. Given sets S_1, S_2 subsets of \mathbb{R}^n , $S_1 + S_2 := \{x_1 + x_2 \mid x_1 \in S_1, x_2 \in S_2\}$. Given a vector $x \in \mathbb{R}^n$, $|x|$ denotes its Euclidean norm. The equivalent notation $[x^\top \ y^\top]^\top$, $[x \ y]^\top$, and (x, y) is used for vectors. Given a function $f : \mathbb{R}^m \rightarrow \mathbb{R}^n$, its domain of definition is denoted by $\text{dom } f$; i.e., $\text{dom } f := \{x \in \mathbb{R}^m \mid f(x) \text{ is defined}\}$. A function $\alpha : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is said to belong to class \mathcal{K}_∞ if it is continuous, zero at zero, strictly increasing, and unbounded. $\mathcal{PC}^0(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$ is the set of all piecewise continuous signals $\beta : \text{dom } \beta \rightarrow \mathbb{R}^m$, $\text{dom } \beta \subset \mathbb{R}_{\geq 0}$.

2.2 Motion planning for nonlinear systems with symmetries

We consider nonlinear control systems of the form

$$\mathcal{P} : \quad \dot{x} = f(x, u) \tag{1}$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a locally Lipschitz function, $x \in \mathbb{R}^n$ is the state, and $u \in \mathbb{R}^m$ is the control input. We focus on a particular subclass of nonlinear systems \mathcal{P} , those satisfying certain symmetry properties. Next, we review and adapt some of the concepts in [6] for the purposes of this paper.

2.2.1 Nonlinear systems with symmetries

A large class of mechanical systems are invariant under certain transformations of their state. These include mobile robots as well as more general autonomous vehicles, like several helicopters and airplanes models, among others. General invariant transformations can be characterized with the theory of Lie groups (see [2] for an introduction to Lie groups and [13] for applications to mechanics).

Let \mathcal{G} be a finite-dimensional Lie group, and let e be its identity element. It is said that Ψ is a left action of the group \mathcal{G} on \mathbb{R}^n if $\Psi : \mathcal{G} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a smooth map such that $\Psi(e, x) = x$ for all $x \in \mathbb{R}^n$ and $\Psi(g, \Psi(h, x)) = \Psi(gh, x)$ for all $g, h \in \mathcal{G}$, $x \in \mathbb{R}^n$. Let \mathfrak{g} be the Lie algebra of \mathcal{G} .

Definition 2.1 (*symmetry of \mathcal{P}*) *The nonlinear system \mathcal{P} is invariant with respect to the left group action Ψ if for all $g \in \mathcal{G}$, $x^0 \in \mathbb{R}^n$, and $\mu \in \mathcal{PC}^0(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$, each solution (in the appropriate sense¹) to \mathcal{P} starting from x^0 with $u(t) = \mu(t)$, denoted by $t \mapsto \phi(x^0, \mu; t)$, is such that $\Psi(g, \phi(x^0, \mu; t)) = \phi(\Psi(g, x^0), \mu; t)$ for all $t \in \text{dom } \phi$. ■*

Definition 2.1 states that \mathcal{P} is invariant if the left action Ψ commutes with the map from initial conditions.

2.2.2 Library of motion primitives

Trim trajectories and maneuvers define our “library” of primitives for motion planning; see also [6, Section III].

Definition 2.2 (trim) *A C^1 function $x : [0, T] \rightarrow \mathbb{R}^n$ is a trim trajectory for \mathcal{P} if there exists $\xi \in \mathfrak{g}$, called the trim velocity vector, and $\mu \in \mathbb{R}^m$, called the trim input, such that*

$$\begin{aligned} x(t) &= \Psi(\exp(\xi t), x(0)) && \text{for all } t \in [0, T], \\ \dot{x}(t) &= f(x(t), \mu) && \text{for almost all } t \in [0, T]. \end{aligned} \quad (2) \quad \blacksquare$$

When the right-hand side of \mathcal{P} is locally Lipschitz, every trim trajectory x for \mathcal{P} is uniquely defined by its velocity ξ and initial condition x^0 . We shall assume the following property throughout the paper.

Standing Assumption 2.3 *The function $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is locally Lipschitz continuous. The nonlinear system \mathcal{P} is invariant under the action of Ψ . ■*

Then, for the nonlinear system \mathcal{P} with symmetry group \mathcal{G} , we store ξ and x^0 in the set of trim trajectories, which is denoted by $\mathcal{T}(\mathcal{P}, \mathcal{G}) \subset \mathfrak{g} \times \mathbb{R}^n$.

¹This property does not depend on the notion of solution used. It is required to hold for each (perhaps nonunique) solution to \mathcal{P} on its domain.

Definition 2.4 (maneuver) A C^1 function $x : [0, T] \rightarrow \mathbb{R}^n$ is a maneuver for \mathcal{P} if there exist a function $\beta \in \mathcal{PC}^0(\mathbb{R}_{\geq 0}, \mathbb{R}^m)$, called the maneuver input, such that

$$\dot{x}(t) = f(x(t), \beta(t)) \quad \text{for almost all } t \in [0, T] ;$$

$g \in \mathcal{G}$, called the maneuver displacement, satisfying

$$x(T) = \Psi(g, x(0)) ;$$

and trim trajectories $x' : [0, T'] \rightarrow \mathbb{R}^n, x'' : [0, T''] \rightarrow \mathbb{R}^n$ that are compatible with x , i.e., there exist matching displacements $g', g'' \in \mathcal{G}$ such that

$$x'(T') = \Psi(g', x(0)), \quad x(T) = \Psi(g'', x''(0)) . \quad \blacksquare$$

Remark 2.5 The matching displacements g' and g'' in Definition 2.4 guarantee that trim trajectories and maneuvers can be concatenated. More precisely, the left action Ψ with displacement g' guarantees that the end point of the (left compatible) trim trajectory x' can be concatenated with the initial point of the maneuver x , while the left action Ψ with displacement g'' guarantees that the initial point of the (right compatible) trim trajectory x'' can be concatenated with the final point of the maneuver x . \blacksquare

Maneuver information for \mathcal{P} with symmetry group \mathcal{G} is stored in the set $\mathcal{M}(\mathcal{P}, \mathcal{G})$. By the regularity properties of f , a maneuver x for \mathcal{P} can be generated by only knowing the input β applied to \mathcal{P} and the initial condition x^0 . By construction, the application of β at x^0 causes a maneuver displacement given by $g \in \mathcal{G}$.

Following the definitions above, a “library” of motion primitives for \mathcal{P} with symmetry group \mathcal{G} is given by $(\mathcal{I}(\mathcal{P}, \mathcal{G}), \mathcal{M}(\mathcal{P}, \mathcal{G}))$. Let $Q_T, Q_M \subset \mathbb{N}$ be compact and disjoint sets, and define $Q := Q_T \cup Q_M$. The set Q_T (respectively, Q_M) is such that each of its elements is uniquely associated to a trim trajectory (respectively, to a maneuver). More precisely, for each $q \in Q_T$, $(\xi_q, x_q^0) \in \mathcal{I}(\mathcal{P}, \mathcal{G})$ defines the trim trajectory $x_q(t) = \Psi(\exp(\xi_q t), x_q^0)$ with $x_q(0) = x_q^0$, while for each $q \in Q_M$, $(\beta_q, x_q^0, g_q, T_q) \in \mathcal{M}(\mathcal{P}, \mathcal{G}) \subset \mathcal{PC}^0(\mathbb{R}_{\geq 0}, \mathbb{R}^m) \times \mathbb{R}^n \times \mathcal{G} \times \mathbb{R}$ correspond to the input to generate the maneuver x_q from x_q^0 , which, after T_q units of time, results in a displacement given by g_q .

2.2.3 Motion plan

A motion plan v is denoted by

$$v := \{(q_1, T_{q_1}), (q_2, g'_2, g''_2), (q_3, T_{q_3}), \dots, \\ \dots, (q_{k-1}, g'_{k-1}, g''_{k-1}), (q_k, T_{q_k})\} ,$$

where $k \in \mathbb{N}_{\geq 3}$ is an odd number and:

- For each odd number $j \in \mathbb{N}_{\leq k}$, $q_j \in Q_T$.
- For each even number $j \in \mathbb{N}_{\leq k}$, $q_j \in Q_M$ and the j -th maneuver is compatible with the $(j-1)$ -th trim trajectory with matching displacement g'_j and with the $(j+1)$ -th trim trajectory with matching displacement g''_j .
- For each odd number $j \in \mathbb{N}_{< k}$, $T_{q_j} \in \mathbb{R}_{\geq 0}$ defines the time to execute the q_j -th trim trajectory. The nonnegative constant T_{q_k} for the last trim trajectory can be either finite or infinite.

In other words, a motion plan v is given by a sequence $\{v_j\}_{j=1}^k$, where v_2, v_4, \dots, v_{k-1} are such that $q_2, q_4, \dots, q_{k-1} \in Q_M$ define maneuvers and v_1, v_3, \dots, v_k are such that $q_1, q_3, \dots, q_k \in Q_T$ define (compatible) trim trajectories. (Alternatively, and without affecting the results in this paper, motion plans can be defined as in [6].) We denote by $\mathcal{V}(\mathcal{P}, \mathcal{G})$ the set of motion plans for \mathcal{P} with symmetry group \mathcal{G} generated from $(\mathcal{T}(\mathcal{P}, \mathcal{G}), \mathcal{M}(\mathcal{P}, \mathcal{G}))$. Figure 1 depicts a sample trim-maneuver-trim piece of a motion plan $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$.

2.3 Hybrid systems

The hybrid control framework proposed in this paper for maneuver-based motion planning follows the general model for hybrid systems in outlined in [8] (see also [9,17]). Hybrid systems are dynamical systems with continuous and discrete dynamics. In [8], a hybrid system \mathcal{H} is given by a flow map, a flow set, a jump map, and a jump set. For the purposes of this paper, the state of the hybrid system, denoted by ζ , takes values in \mathbb{R}^n , the flow map is given by a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and the flow set, denoted by $C \subset \mathbb{R}^n$, define the flow equation $\dot{x} = f(x), x \in C$; while the jump map is given by a function $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and the jump set, denoted by $D \subset \mathbb{R}^n$, define the jump equation $x^+ = g(x), x \in D$. Continuous evolution of the solutions (or flows)

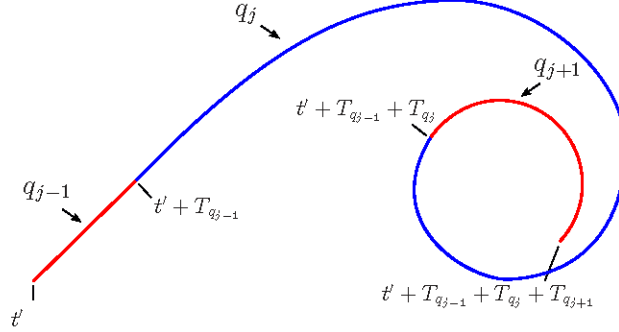


Figure 1: Sequence of entries of a motion plan v : $v_{j-1} = (q_{j-1}, T_{q_{j-1}})$ defining trim trajectory $x_{q_{j-1}}$, $v_j = (q_j, g'_{q_j}, g''_{q_j})$ defining maneuver x_{q_j} , and $v_{j+1} = (q_{j+1}, T_{q_{j+1}})$ defining trim trajectory $x_{q_{j+1}}$.

to \mathcal{H} is permitted only when the solution is in C and discrete evolution (or jumps) is allowed only when the solution is in D . Hence, a hybrid system \mathcal{H} has data (f, C, g, D) and can be written as

$$\mathcal{H} : \quad x \in \mathbb{R}^n \quad \begin{cases} \dot{x} &= f(x), & x \in C \\ x^+ &= g(x), & x \in D. \end{cases}$$

To define solutions to \mathcal{H} , the number of jumps is treated as an independent variable j and the state is parametrized by (t, j) . A solution is a function defined on subsets of $\mathbb{R}_{\geq 0} \times \mathbb{N}$. A subset $E \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$ is a *compact hybrid time domain* if

$$E = \bigcup_{j=0}^{J-1} ([t_j, t_{j+1}], j)$$

for some finite sequence of times $0 = t_0 \leq t_1 \leq \dots \leq t_J$. It is a *hybrid time domain* if for all $(T, J) \in E$, $E \cap ([0, T] \times \{0, 1, \dots, J\})$ is a compact hybrid domain. On each hybrid time domain there is a natural ordering of points: $(t, j) \preceq (t', j')$ if $t \leq t'$ and $j \leq j'$. A *hybrid arc* is a function $x : \text{dom } x \rightarrow \mathbb{R}^n$ on a hybrid time domain $\text{dom } x$ such that $x(t, j)$ is absolutely continuous in t for a fixed j and $(t, j) \in \text{dom } x$. It is a solution to the hybrid system \mathcal{H} if $x(0, 0) \in \overline{C} \cup D$ and

$$(S1) \text{ For all } j \in \mathbb{N} \text{ and almost all } t \text{ such that } (t, j) \in \text{dom } x, \\ x(t, j) \in C, \quad \dot{x}(t, j) = f(x(t, j))$$

$$(S2) \text{ For all } (t, j) \in \text{dom } x \text{ such that } (t, j+1) \in \text{dom } x, \\ x(t, j) \in D, \quad x(t, j+1) = g(x(t, j)).$$

A concept of closeness of solutions to hybrid systems is as follows. Two solutions $x : \text{dom } x \rightarrow \mathbb{R}^n$, $y : \text{dom } y \rightarrow \mathbb{R}^n$ are (T, J, ε) -close if:

- (a) for all $(t, j) \in \text{dom } x$ with $t \leq T$, $j \leq J$ there exists s such that $(s, j) \in \text{dom } y$, $|t - s| < \varepsilon$, and

$$|x(t, j) - y(s, j)| < \varepsilon,$$

- (b) for all $(t, j) \in \text{dom } y$ with $t \leq T$, $j \leq J$ there exists s such that $(s, j) \in \text{dom } x$, $|t - s| < \varepsilon$, and

$$|y(t, j) - x(s, j)| < \varepsilon.$$

Note that this closeness concept does not require solutions to be close at jumps at the same hybrid instant (t, j) . See [8] and [9] for more details.

3 A hybrid controller for motion planning

Given a motion plan $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$, our goal is to design a controller generating a trajectory of \mathcal{P} that satisfies the motion plan specifications given in terms of a finite sequence of trim trajectories and maneuvers from $(\mathcal{T}(\mathcal{P}, \mathcal{G}), \mathcal{M}(\mathcal{P}, \mathcal{G}))$. We propose a hybrid controller, denoted by \mathcal{H}_c , with:

- logic state $q \in Q$ to indicate the system mode: *trim mode* when $q \in Q_T$, *maneuver mode* when $q \in Q_M$.
- logic state $p \in \mathbb{N}$ to select an entry of a given motion plan $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$.
- displacement state $z \in \mathcal{G}$ to store the overall displacement of the trajectory of \mathcal{P} .
- timer state $\tau \in \mathbb{R}$ to keep track of the time in maneuver mode and to parametrize the reference trajectory during trim mode.

The output of the controller, that is, the input of \mathcal{P} , is

$$u = \kappa_c(x, q, \tau) \tag{3}$$

where $\kappa_c : \mathbb{R}^n \times Q \times \mathbb{R} \rightarrow \mathbb{R}^m$. The input to \mathcal{H}_c is the state x of \mathcal{P} .

3.1 Control strategy

Given a motion plan $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$, let $q = q_j \in Q_T$, $j \in \mathbb{N}_{<k}$. The controller \mathcal{H}_c performs the following tasks:

Task 1) *Trim Trajectory Tracking*: Track the trim trajectory x_q , where x_q is defined by $(\xi_q, x_q^0) \in \mathcal{T}(\mathcal{P}, \mathcal{G})$ via (2).

Task 2) *Maneuver Execution Start*: When the state x is such that the maneuver $x_{q_{j+1}}$, which succeeds the trim trajectory x_q , can be executed and the timer elapsed for at least T_q units of time, update q to q_{j+1} , reset timer τ to zero, and execute the $(j+1)$ -th maneuver.

Task 3) *Maneuver Execution End*: When the state x is such that the trim trajectory $x_{q_{j+2}}$ can be executed and the timer τ has elapsed for at least T_q units of time, update q to q_{j+2} and perform Task 1) if $j+2 < k$.

Execution of trim trajectories in Task 1 is performed in closed-loop with a local tracking controller that guarantees $x(t) \rightarrow x_q(t)$ asymptotically. Maneuvers are started when: 1) the timer has elapsed for at least the duration planned for the predecessor trim trajectory, and 2) the state reaches a set from where the maneuver can be executed (the latter corresponds to Task 2). The trim trajectory that follows every maneuver is started as soon as the state x is in the set where tracking is possible and the timer has elapsed the specified amount of time for the maneuver.

3.2 Control design

The following assumption guarantees that Task 1 can be accomplished.

Assumption 3.1 (*tracking of trim trajectories*) *For each $q \in Q_T$, there exists a continuous function $\kappa_q : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^m$, a continuously differentiable function $V_q : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$, class- \mathcal{K}_∞ functions α_q^1, α_q^2 , and an open neighborhood of the origin $\mathcal{B}_q \subset \mathbb{R}^n$ such that*

$$\begin{aligned} \alpha_q^1(|e|) &\leq V_q(e) \leq \alpha_q^2(|e|) & \forall e \in \mathbb{R}^n, \\ \langle V_q(e), \tilde{f}(e) \rangle &\leq -V_q(e) & \forall e \in \mathcal{B}_q, \end{aligned} \quad (4)$$

where $\tilde{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is given by

$$\tilde{f}(e) = f(e + x_q(t), \kappa_q(e + x_q(t), t) - f(x_q(t), \mu_q)) ,$$

and defines the time-invariant system $\dot{e} = \tilde{f}(e)$ invariant under the action of Ψ , where x_q is the trim trajectory generated by μ_q .

Remark 3.2 *In addition to the invariance property, Assumption 3.1 guarantees the existence of a local controller, with basin of attraction \mathcal{B}_q , which accomplishes asymptotic tracking of trim trajectories. Additionally, each tracking control law κ_q is such that, when applied to \mathcal{P} , result in a time-invariant error system with $e := x - x_q$ having the symmetry property. This assumption holds for nonlinear systems that can be put in feedback linearizable normal form [3, 10] with error system that is invariant under the action of Ψ [15].*

■

The construction of the flow and jumps sets of \mathcal{H}_c follows. By the continuity properties of maneuvers in Definition 2.4, for each maneuver x_q with input β_q and maneuver duration T_q , $q \in Q_M$, there exist disjoint and open sets $S_q, L_q \subset \mathbb{R}^n$ such that for each $x_q(0) \in S_q$, $x_q(T_q) \in L_q$, $\dot{x}_q(t) = f(x_q, \beta_q(t))$. For each $q \in Q_M$, pick compact sets D_q such that $D_q \subset S_q$ and $x_q^0 \in D_q^\circ$, and define $C_q := \mathbb{R}^n \setminus \overline{D_q}$. The set D_q , $q \in Q_M$, corresponds to the maneuver's start set in Task 2.

We now compute the set of points from where tracking of trim trajectories is possible. By construction, there exist $\varepsilon^* > 0$ such that

$$\varepsilon^* := \operatorname{argmax}_{\varepsilon > 0} \{x_q^0 + \varepsilon \overline{\mathbb{B}} \subset S_q, \forall q \in Q_M\} .$$

Using Assumption 3.1, for each $q \in Q_T$, define

$$D_q := \{e \in \mathbb{R}^n \mid V_q(e) \leq c_q\} ,$$

where $c_q > 0$ is such that

$$D_q \subset (x_q^0 + \delta_q \overline{\mathbb{B}}) \cap \mathcal{B}_q , \quad \delta_q := (\alpha_1^q)^{-1}(\exp(T_q)\alpha_2^q(\varepsilon^*)) ,$$

and $(\alpha_1^q)^{-1}$ is the inverse of the function α_1^q . Define $C_q := \mathbb{R}^n \setminus \overline{D_q}$. This construction yields a constant δ_q such that when the trim trajectory $x_q(t)$ is tracked from initial conditions in D_q , the state x belongs to a subset of the start set of each of the maneuvers in $\mathcal{M}(\mathcal{P}, \mathcal{G})$ after T_q units of time have elapsed (T_q is the execution time of the trim trajectory given in the motion plan).

The following assumption guarantees that maneuvers take trajectories to points where trim trajectories can be executed.

Assumption 3.3 (nested condition) For every motion plan $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$, every maneuver with associated entry v_i in v and input β_{q_i} , its associated set L_{q_i} is such that

$$L_{q_i} \subset D_{q_{i+1}},$$

where $D_{q_{i+1}}$ is the set associated with tracking of the trim trajectory $x_{q_{i+1}}$, $q_{i+1} \in Q_T$.

Remark 3.4 The condition in Assumption 3.3 assures that, after a maneuver, the state x is in a set from which tracking of the trim trajectory succeeding it is possible. This condition holds by picking small enough landing set L_q when Assumption 3.1 is in place. However, in order to get practical robustness results, the landing sets are usually fixed. In such cases, the tracking law in Assumption 3.1 should be chosen to have large enough set $D_q, q \in Q_T$. ■

Figure 2 illustrates the sets designed above.

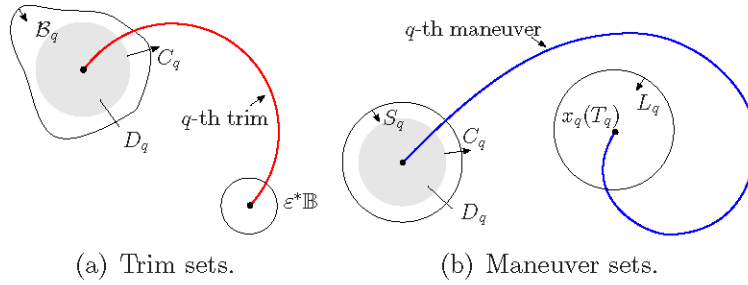


Figure 2: Sets of the hybrid controller for a trim trajectory and maneuver in the motion primitive in Figure 1.

3.3 Hybrid controller

The control logic outlined above is implemented in the hybrid controller \mathcal{H}_c as follows.

3.3.1 Jumps

Jumps occur while in trim mode with $p < k$ (i.e., it is not the last trim trajectory of the motion plan) when the state x reaches the set of points where the maneuver $x_{q_{p+1}}$ can be started and the timer τ has elapsed for T_{q_p} units of time. The set in the first condition is given by $D_{q_{p+1}}$, $q_{p+1} \in Q_M$, after the left action Ψ with displacement given by z multiplied by the nominally expected trim trajectory displacement $\exp(\xi_q T_q)$ and the matching displacement $g'_{q_{p+1}}$. Then, jumps occur when

$$q \in Q_T \text{ and } x \in \Psi(z \exp(\xi_q T_q) g'_{q_{p+1}}, D_{q_{p+1}}) \text{ and } \tau \geq T_q, \quad (5)$$

with update law

$$q^+ = q_{p+1}, \quad p^+ = p + 1, \quad z^+ = z \exp(\xi_q \tau), \quad \tau^+ = 0, \quad (6)$$

that is, q is mapped to the next mode in the motion plan v , the motion plan index p is incremented by one, z is updated with the current total displacement of the motion primitive, and τ is reset to zero.

While in maneuver mode, jumps occur when the state reaches the set of points where the trim trajectory $x_{q_{p+1}}$ can be started and the timer state τ has elapsed for at least T_q units of time. As in the case for jumps during trim mode, the set in the former condition is given by D_q , $q \in Q_M$, after the invariant operation Ψ with displacement given by z multiplied by the planned maneuver trajectory displacement, which is given by g_q , and the matching displacement g''_q . Then, jumps in maneuver mode occur when

$$q \in Q_M \text{ and } x \in \Psi(z g_q g''_q, D_{q_{p+1}}) \text{ and } \tau \geq T_q, \quad (7)$$

with update law

$$q^+ = q_{p+1}, \quad p^+ = p + 1, \quad z^+ = z g_q, \quad \tau^+ = 0. \quad (8)$$

3.3.2 Flows

During flows, the controller variables have dynamics given by

$$\dot{q} = 0, \quad \dot{p} = 0, \quad \dot{z} = 0, \quad \dot{\tau} = 1, \quad (9)$$

when

$$q \in Q_T \text{ and } (x \in \Psi(z \exp(\xi_q T_q) g'_{q_{p+1}}, C_{q_{p+1}}) \text{ or } \tau \in [0, T_q]), \quad (10)$$

or

$$q \in Q_M \text{ and } (x \in \Psi(z g_q g''_q, C_{q_{p+1}}) \text{ or } \tau \in [0, T_q]). \quad (11)$$

3.3.3 Output

The controller output is the input to \mathcal{P} and is given by $u = \kappa_c(x, q, \tau)$ where

$$\kappa_c(x, q, \tau) = \begin{cases} \beta_q(\tau) & \text{if } q \in Q_M \\ \kappa_q(x, \tau) & \text{if } q \in Q_T. \end{cases} \quad (12)$$

The function β_q is the control input that generates the q -th maneuver, $q \in Q_M$. The function κ_q is the tracking control law in Assumption 3.1 for the q -th trim trajectory, $q \in Q_T$, which is designed using trim trajectory information.

3.3.4 Closed-loop system

We denote the closed-loop system resulting from controlling \mathcal{P} with \mathcal{H}_c by \mathcal{H}_{cl} and its state by $\varphi := (x, q, p, z, \tau) \in \mathcal{X} := \mathbb{R}^n \times Q \times \mathbb{N}_{\leq k} \times \mathbb{R}^l \times \mathbb{R}$, where the Euclidean space \mathbb{R}^l embeds \mathcal{G} . The continuous dynamics are given by closed-loop plant dynamics $\dot{x} = f(x, \kappa_c(x, q, \tau))$ along with (9), with flow set given by the union of the sets defined by (10) and (11). The discrete dynamics are given by the update laws in (6) and (8). The resulting closed-loop system \mathcal{H}_{cl} can be written in the compact form in (2.3) using φ as the state and appropriately defining functions \tilde{f}, \tilde{g} and sets \tilde{C} and \tilde{D} .

4 Motion plan execution: nominal and perturbed case

Given a motion plan v and an initial configuration $(x_v^0, g_v^0) \in \mathbb{R}^n \times \mathcal{G}$ such that $x_v^0 = \Psi(g_v^0, x_{q_1}^0)$, let $r : \text{dom } r \rightarrow \mathbb{R}^n$ describe the desired trajectory of

the nominal motion plan v , that is:

$$r(t, j) = \begin{cases} \Psi(g_v^0, x_{q_1}(t)) & \text{if } t \in [0, T_1], \\ & \text{and } j = 0 \\ \Psi(g_v^0 \exp(\xi_{q_1} T_1) g'_1, x_{q_2}(t)) & \text{if } t \in [T_1, T_2], \\ & \text{and } j = 1 \\ \Psi(g_v^0 \exp(\xi_{q_1} T_1) g''_1, x_{q_3}(t)) & \text{if } t \in [T_2, T_3], \\ & \text{and } j = 2 \\ \vdots & \vdots \\ \Psi(g_v^0 \exp(\xi_{q_1} T_1) g''_1 \dots & \text{if} \\ \dots \exp(\xi_{q_{k-1}} T_{k-1}) g''_k, x_{q_k}(t)) & t \in [T_{k-1}, T_k], \\ & \text{and } j = k, \end{cases}$$

where x_{q_1} is the trim trajectory with $(\xi_{q_1}, x_{q_1}^0) \in \mathcal{T}(\mathcal{P}, \mathcal{G})$, x_{q_2} is the maneuver with $(\beta_{q_2}, x_{q_2}^0, g_{q_2}, T_{q_2}) \in \mathcal{M}(\mathcal{P}, \mathcal{G})$, etc. Note that each jump of r corresponds to a change of motion primitive. For example, for each $(t, j) \in [0, T_1] \times \{0\}$, $r(t, j)$ is given by the q_1 -th trim trajectory, and after the jump at $t = T_1, j = 0$, and for all $(t, j) \in [T_1, T_2] \times \{1\}$, $r(t, j)$ is given by the q_2 -th maneuver. The duration of the motion plan v is $T_r = \sum_{i=1,3,\dots,k} T_i + \sum_{i=2,4,\dots,k-1} T_{q_i}$. When T_r is finite, $\text{dom } r$ is a subset of $[0, T_r] \times \{0, 1, 2, \dots, k-1\}$, while when T_r is infinite, $\text{dom } r$ is a subset of $[0, \infty) \times \{0, 1, 2, \dots, k-1\}$.

Theorem 4.1 (*nominal execution*) *Let Assumptions 3.1 and 3.3 hold. For each $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$ with nominal motion plan trajectory r and each $(x_v^0, g_v^0) \in \mathbb{R}^n \times \mathcal{G}$ such that $x_v^0 = \Psi(g_v^0, x_{q_1}^0)$, $(\xi_{q_1}, x_{q_1}^0) \in \mathcal{T}(\mathcal{P}, \mathcal{G})$, there exists a unique solution φ to \mathcal{H}_{cl} from $\varphi(0, 0) = (x_v^0, q_1, 1, g_v^0, 0)$ that is bounded and is such that the x component satisfies $x(t, j) = r(t, j)$ for all $(t, j) \in \text{dom } \varphi$.*

Remark 4.2 *Theorem 4.1, which follows by construction, states that every motion plan $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$ is properly executed by \mathcal{H}_{cl} . This result recovers the nominal motion plan execution property of the hybrid automaton in [6].* ■

In addition to the nominal property in Theorem 4.1, the proposed hybrid control construction guarantees that, under the presence of perturbations, motion plan execution stay close to a nominal one. Note that the presence of perturbations in \mathcal{H}_{cl} on the initial conditions, parameters, and/or the state affects the jump times. In this way, the domain of the resulting trajectory

does not need to coincide with the domain of the nominal trajectory r associated to $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$. The (T, J, ε) -closeness notion of distance between hybrid arcs in Section 2.3 handles such a situation.

Theorem 4.3 (*perturbation of initial conditions*) *Let Assumptions 3.1 and 3.3 hold. For each $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$ with nominal motion plan trajectory r and each $(x_v^0, g_v^0) \in \mathbb{R}^n \times \mathcal{G}$ such that $x_v^0 = \Psi(g_v^0, x_{q_1}^0)$, $(\xi_{q_1}, x_{q_1}^0) \in \mathcal{T}(\mathcal{P}, \mathcal{G})$, each $\varepsilon > 0$, each compact set $K \subset \mathcal{B}_{q_1}$, and each $(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$, $(T, J) \preceq (T_r, k-1)$, there exists $\delta > 0$ such that every solution φ_δ to \mathcal{H}_{cl} with $\varphi_\delta(0, 0) = (x_\delta^0, q_1, 1, g_v^0, 0)$, $x_\delta^0 \in K + \delta\mathbb{B}$, is bounded and the x component and r are (T, J, ε) -close.*

Remark 4.4 *The time horizon (T, J) where the closeness property in Theorem 4.3 holds can be picked to be equal to $(T_r, k-1)$ when T_r is finite. Then, closeness between the component x of the solution and r is guaranteed in the entire duration of the motion plan. The hybrid time domain of each solution to \mathcal{H}_{cl} can be extended to an unbounded one without affecting the behavior of the system up to time (T, J) . In addition to the regularity properties of the closed-loop system (guaranteed by the standing assumption and the hybrid controller construction), the proof of Theorem 4.3 extends the hybrid time domain to an unbounded one to enable the application of results in [9] for hybrid systems with perturbations. ■*

Under the presence of perturbations, system \mathcal{P} controlled by \mathcal{H} can be written as

$$\dot{x} = f(x, \kappa_c(x + d_1(t), q, \tau)) + d_2(t) , \quad (13)$$

where d_1 corresponds to error in the measurements of x and d_2 models other exogenous disturbances and unmodeled dynamics. The addition of these perturbations in the closed-loop system \mathcal{H}_{cl} results in a perturbed hybrid system, denote as $\tilde{\mathcal{H}}_{cl}$, which can be written as

$$\begin{aligned} \dot{\varphi} &= \tilde{f}(\varphi + d_1(t)) + d_2(t) & \varphi + d_1 &\in \tilde{C} \\ \varphi^+ &= \tilde{g}(\varphi) & \varphi + d_1 &\in \tilde{D} , \end{aligned}$$

The following result asserts that the motion planning is robust to a class of perturbations.²

²The exogenous signals d_1 and d_2 are given on hybrid time domains (given a hybrid time domain S and an exogenous signal $d_1(t)$, we can define, with some abuse of notation, $d_1(t, j) := d_1(t)$ for each $(t, j) \in S$.) Solutions to hybrid systems with the perturbations above is understood similarly to the notion of solution outlined in Section 2.3.

Theorem 4.5 (*perturbations*) *Let Assumptions 3.1 and 3.3 hold. For each $v \in \mathcal{V}(\mathcal{P}, \mathcal{G})$ with nominal motion plan trajectory r and each $(x_v^0, g_v^0) \in \mathbb{R}^n \times \mathcal{G}$ such that $x_v^0 = \Psi(g_v^0, x_{q_1}^0)$, $(\xi_{q_1}, x_{q_1}^0) \in \mathcal{T}(\mathcal{P}, \mathcal{G})$, each $\varepsilon > 0$, each compact set $K \subset \mathcal{B}_{q_1}$, and each $(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$, $(T, J) \preceq (T_r, k-1)$, there exists $\delta > 0$ such that every solution $\tilde{\varphi}$ to \mathcal{H}_{cl} with $\tilde{\varphi}(0, 0) = (x^0, q_1, 1, g_v^0, 0)$, $x^0 \in K + \delta\mathbb{B}$, $|d_1(t, j)| \leq \delta$ and $|d_2(t, j)| \leq \delta$ for each $(t, j) \in \text{dom } \varphi$, is bounded and the x component and the motion plan trajectory r are (T, J, ε) -close.*

Remark 4.6 *The proof of this result uses a technique from [9, Section V] in which a perturbed hybrid system \mathcal{H}_{cl}^δ is embedded into a set-valued hybrid system. Using the hybrid time domain extension as in Theorem 4.3, the results follows from [9, Corollary 5.5].* ■

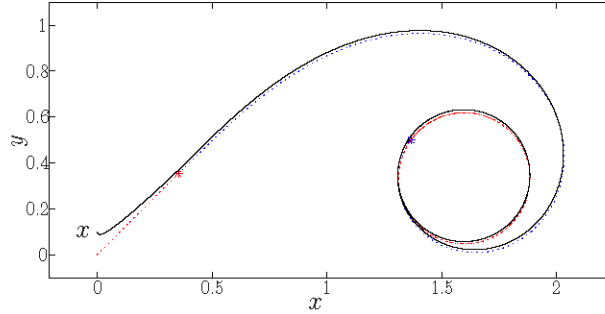


Figure 3: Motion primitive (dashed) in Figure 1 and simple airplane trajectory resulting from applying our hybrid control strategy for motion planning. Tracking control during trims (red pieces) guarantees that solution and trim trajectory are stay close. Maneuver starts from a point nearby the maneuver (blue piece) in the library and remains close to it.

Finally, Figure 3 illustrates a solution to \mathcal{H}_{cl} starting nearby the motion plan in Figure 1. This corresponds to a simulation result from a toolbox for robust maneuver-based motion planning, currently under development.

5 Conclusion

We presented a hybrid systems framework for maneuver-based motion planning algorithms for nonlinear systems with symmetries. We systematically described the construction of a hybrid controller and showed its robustness properties for a large class of perturbations. Our results are built upon recent tools for robustness of stability for hybrid systems. Extensions of the hybrid control strategy to situations where bounds on the perturbations are known beforehand follow from the ideas presented in this manuscript and will be closely explored in the future.

References

- [1] M. Brady. *Robot Motion: Planning and Control*. Mit Press, 1982.
- [2] C. C. Chevalley. *Theory of Lie Groups*. Princeton Univ Press, 1946.
- [3] F. Esfandiari and H. K. Khalil. Output feedback stabilization of fully linearizable systems. *International Journal of Control*, 56:1007–1037, 1992.
- [4] E. Frazzoli, M. A. Dahleh, and E. Feron. Robust hybrid control for autonomous vehicle motion planning. *Proc. 39th IEEE Conference on Decision and Control*, 1:821–826, 2000.
- [5] E. Frazzoli, M. A. Dahleh, and E. Feron. Real-time motion planning for agile autonomous vehicles. *Journal of Guidance, Control, and Dynamics*, 25:116–129, 2002.
- [6] E. Frazzoli, M. A. Dahleh, and E. Feron. Maneuver-based motion planning for nonlinear systems with symmetries. *Robotics, IEEE Transactions on [see also Robotics and Automation, IEEE Transactions on]*, 21:1077–1091, 2005.
- [7] V. Gavrillets, E. Frazzoli, B. Mettler, M. Piedmonte, and E. Feron. Aggressive maneuvering of small autonomous helicopters: A human-centered approach. *The International Journal of Robotics Research*, 20:795, 2001.

- [8] R. Goebel, J.P. Hespanha, A.R. Teel, C. Cai, and R.G. Sanfelice. Hybrid systems: generalized solutions and robust stability. In *Proc. 6th IFAC Symposium in Nonlinear Control Systems*, pages 1–12, 2004.
- [9] R. Goebel and A.R. Teel. Solutions to hybrid inclusions via set and graphical convergence with stability theory applications. *Automatica*, 42(4):573–587, 2006.
- [10] A. Isidori. *Nonlinear control systems: an introduction*. Springer-Verlag New York, Inc. New York, NY, USA, 1989.
- [11] J.-P. Laumond, editor. *Robot Motion Planning and Control*, volume 229 of *Lectures Notes in Control and Information Sciences*. Springer Verlag, 1998.
- [12] S. M. LaValle. *Planning algorithms*. Cambridge University Press, Cambridge, UK, 2006.
- [13] J. E. Marsden and T. S. Ratiu. *Introduction to mechanics and symmetry*. Springer New York, 1999.
- [14] A. E. Quaid and A. A. Rizzi. Robust and efficient motion planning for a planar robot using hybrid control. In *Proc. ICRA*, 2000.
- [15] P. Rouchon and J. Rudolph. Invariant tracking and stabilization: Problem formulation and examples. *Lecture notes in control and information sciences*, 246:261–273, 1999.
- [16] R. G. Sanfelice and A. R. Teel. A “throw-and-catch” hybrid control strategy for robust global stabilization of nonlinear systems. In *Proc. 26th American Control Conference*, pages 3470–3475, 2007.
- [17] R.G. Sanfelice, R. Goebel, and A.R. Teel. Generalized solutions to hybrid dynamical systems. *ESAIM: Control, Optimisation and Calculus of Variations*, 2008.
- [18] T. Schouwenaars, B. Mettler, E. Feron, and J.P. How. Robust motion planning using a maneuver automation with built-in uncertainties. In B. Mettler, editor, *Proc. Amer. Control Conf.*, pages 2211–2216, 2003.

D Appendix:
Discussion on “Optimality Properties and
Driver Input Parameterization for Trail-
Braking Cornering”

Discussion on “Optimality Properties and Driver Input Parameterization for Trail-Braking Cornering”

Emilio Frazzoli

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology

1 Overview

The paper by Velenis, Tsiotras, and Lu [6] is a very interesting contribution towards the development of automatic control systems able to push the performance envelopes of autonomous vehicles and mobile robots. In fact, while automatic control techniques routinely show superior performance (with respect to humans), e.g., in terms of set-point regulation, the dexterity and nimbleness demonstrated by human-operated vehicles and machines is as yet largely unattainable by automated systems.

Advances towards the development of autonomous aircraft capable of performing human-inspired acrobatic maneuvers have been reported, e.g., in [4, 5]. The work of Velenis *et al.* is a first step in the context of Ackermann-steered (i.e., car-like) ground vehicles, with a special emphasis on techniques applicable to driving on loose terrain.

Both [4] and [6] follow a similar basic process: (i) data are collected from an instrumented vehicle, while an expert human pilot executes a maneuver of interest; (ii) these data are used to validate an analytical model of the vehicle’s dynamics, applicable to the operating conditions encountered during the maneuver; (iii) the data are further interpreted to parameterize and design a control strategy amenable to implementation on a model-based control strategy; (iv) such control strategy is finally assessed on a high-fidelity simulation.

A significant difference (aside from the application domain) is that Velenis *et al.* investigate the optimality of the maneuver, and develop a parameterization of control strategies to reduce the search space of a nonlinear programming algorithm. This enables them to extend the applicability of the maneuver beyond the original scenario: for example, they are able to compute trail-braking maneuvers through a *variety* of corners, with different total turning angles.

2 Towards a symbolic approach to autonomous high-speed driving

As argued in [6], the development of methods for the design of control laws to perform a certain class of maneuver is motivated by the prospect of building a library of such maneuvers, and then stringing together such maneuvers in such a way to construct more complicated trajectories. This is the basis concept behind a promising new research direction in the literature on robotics and automatic control, which is often referred to as *symbolic control*; see, e.g., [2] for a general introduction.

A formal approach to the intuitive concept described above was presented in [3]. The key property of a dynamical system enabling such an approach is *symmetry*, i.e., invariance to a certain class of transformations on the state of the system. This is a very general property of man-made vehicles: for example, the dynamics of a car-like vehicle, operating on flat, horizontal terrain, are invariant with respect to rigid-body motions on the horizontal plane. The existence of symmetries in a dynamical system allows the definition of so-called “motion primitives,” i.e., equivalence classes of trajectories modulo the symmetry transformations.

In [3], in order to develop a systematic approach to the selection of motion primitives to include in a library, two kinds of motion primitives were identified: trim trajectories and maneuvers. The former, also called relative equilibria or steady-state trajectories, correspond to orbits of the infinitesimal action of the symmetry transformation: for example, in the case of a car-like vehicle, such trim trajectories correspond to circles described at constant speed, steering angle, and throttle settings (this includes degenerate circles, such as straight lines). Maneuvers are then defined as transitions between such trim trajectories. The approach in [6] can in fact be seen as a way to compute optimal maneuvers: in particular, the trail-braking maneuver described in the paper can be thought of as a transition between a trim trajectory in which the car is moving straight at constant (high) speed, back to the same trim trajectory, with a different initial point, and a different heading.

The objective of the remainder of this note is complementary to [6]: instead of computing maneuvers, we will analyze trim trajectories for the same dynamic model. The purpose of this investigation is not only to provide well-defined “starting” and “ending” states for trail-braking (or other) maneuvers, but also to investigate possible ways to further decompose such maneuvers. For example, one could imagine decomposing a trail-braking maneuver into (i) an entry phase, in which the car brakes hard and then steers to enter a circular trajectory at a high sideslip angle; (ii) a steady-state phase, in which the car makes progress around the corner, on a tight circle; and (iii) an exit phase, in which the car countersteers while accelerating out of the turn. Varying the length of the “trim” phase, the car would be able to execute a whole class of trail-braking turns, through a range of angles—without the need to compute explicitly several different trajectories.

3 Trim trajectories for the half-car model

Recall that since the dynamics of the car are invariant to rigid body motions in the plane, trim trajectories are (arcs) of circles, followed at constant speed. It is convenient to rewrite the half-car dynamics equations in [6] in a reference frame whose origin is moving on a circle of radius r at constant speed V , rotating at angular velocity $\omega = V/r$, and with (centripetal) acceleration $a = \omega^2 r$. The x axis of this frame is aligned with the velocity vector. Using the same notation as in [6], we get

$$\begin{aligned} m(\ddot{x} - 2\omega\dot{y} - \omega^2 x) &= f_{Fx} \cos(\psi + \delta) - f_{Fy} \sin(\psi + \delta) + f_{Rx} \cos \psi - f_{Ry} \sin \psi, \\ m(\ddot{y} + 2\omega\dot{x} - \omega^2 y + \omega^2 r) &= f_{Fx} \sin(\psi + \delta) + f_{Fy} \cos(\psi + \delta) + f_{Rx} \sin \psi + f_{Ry} \cos \psi, \\ I_z \ddot{\psi} &= (f_{Fx} \sin \delta + f_{Fy} \cos \delta) \ell_F - f_{Ry} \ell_R. \end{aligned} \quad (1)$$

For the motion of the vehicle to remain planar, the following balance equations, and constraint force inequalities, must be satisfied:

$$\begin{aligned} 0 &= f_{Fz} + f_{Rz} - mg, \\ 0 &= h(f_{Fx} \cos \delta - f_{Fy} \sin \delta + f_{Rx}) + f_{Fz} \ell_F - f_{Rz} \ell_R, \\ 0 &= M_x + h(f_{Fx} \sin \delta + f_{Fy} \cos \delta + f_{Ry}), \end{aligned} \quad (2)$$

$$f_{Fz} \geq 0, \quad f_{Rz} \geq 0, \quad |M_x| \leq mg \frac{w}{2}. \quad (3)$$

(Note that the first two constraints (3) are usually satisfied for full-size vehicles. The third one—in which w is the width of the wheelbase, and M_x is the reaction moment along the vehicle’s longitudinal axis—could be violated for large lateral accelerations occurring in vehicles with $w/2h > 1$ (e.g., trucks or SUV’s) and indicates the onset of roll-over phenomena.)

The friction forces f_{Fx} , f_{Fy} , f_{Rx} , and f_{Ry} can be found multiplying the constraint forces f_{Fz} and f_{Rz} by an appropriate friction coefficient, which in turns depends on the amount of slippage between the tire and the terrain. A popular model is Pacejka’s “Magic Formula” [1], which can be written in its basic form as

$$\mu(s) = D \sin(C \arctan(Bs)), \quad (4)$$

where B , C , and D are appropriate constants, and s measures the slip ratio.

For the purpose of this note, as in [6], the slip ratio is defined as the ratio between the relative speed of the wheel and of the terrain at the contact point, and the speed of the contact point on the wheel in the absence of driving/braking torques. With this definition in mind, we obtain

$$s = \sqrt{s_x^2 + (\tan \alpha)^2},$$

where α is the wheel slip angle, formed by the longitudinal axis of the wheel with the velocity vector at the contact point, and s_x is the longitudinal slip, defined as

$$s_x = \frac{\omega_{\text{wheel}} r_{\text{wheel}}}{V \cos \alpha} - 1.$$

Lateral and longitudinal friction coefficients are recovered as

$$\begin{aligned}\mu_x(s_x, s) &= \frac{s_x}{s} \mu(s), \\ \mu_y(\alpha, s) &= \frac{\tan \alpha}{s} \mu(s),\end{aligned}$$

and finally we obtain the lateral and longitudinal friction force components as $f_x = \mu_x f_z$, $f_y = \mu_y f_z$

For simplicity, we will consider the front and rear longitudinal slip ratios s_{Fx} and s_{Rx} as independent control inputs (in place of, e.g., torques on the wheel).

3.1 Computing steady-state trajectories

Steady-state trajectories correspond to equilibrium points in the rotating frame, and can be found solving the system of equations obtained from (1), (2), and (3), setting x , y , ψ , and their derivatives to zero.

Some algebra leads to

$$f_{Ry} = m \omega^2 r \cos \psi \frac{\ell_F}{\ell_F + \ell_R}, \quad (5)$$

$$f_{Rz} = m \frac{g \ell_F + h \omega^2 r \sin \psi}{(\ell_F + \ell_R)}, \quad (6)$$

and

$$f_{Fz} = m \frac{g \ell_R - h \omega^2 r \sin \psi}{(\ell_F + \ell_R)}.$$

Combining (5) and (6), we get that the rear-wheel lateral friction coefficient μ_{Ry} is

$$\mu_{Ry} = \frac{f_{Ry}}{f_{Rz}} = \frac{\omega^2 r \ell_F \cos \psi}{g \ell_R + h \omega^2 r \sin \psi}. \quad (7)$$

Since by definition

$$\mu_{Ry} = \frac{\tan \alpha_R}{s_R} \mu(s_R), \quad (8)$$

equations (7) and (8) can be solved for s_R as a function of the vehicle's linear and angular velocity V and ω , and of the sideslip angle ψ . In particular, if we consider the case of a front-wheel drive (FWD) vehicle, and assume that it is not braking, then $s_{Rx} = 0$ and hence $s_R = \tan \alpha_R$. In this case, (7) and (8) simplify to

$$\frac{\omega^2 r \ell_F \cos \psi}{g \ell_R + h \omega^2 r \sin \psi} = \mu(\tan \alpha_R). \quad (9)$$

The total front-wheel slip ratio can be computed by solving

$$\mu(s_F) = \frac{\sqrt{f_{Fx}^2 + f_{Fy}^2}}{f_{Fz}} = \frac{\sqrt{m^2 \omega^4 r^2 + f_{Ry}^2 - 2m\omega r f_{Ry} \cos \psi}}{f_{Fz}} \quad (10)$$

for s_F . The steering angle δ necessary to achieve the above slip ratio can be found as a solution of the following equation:

$$\frac{\tan(\delta - \delta_0)}{s_F} = \frac{m \omega^2 r \cos(\psi + \delta) + f_{Ry} \cos \delta}{\sqrt{f_{Fx}^2 + f_{Fy}^2}}, \quad (11)$$

where δ_0 is the steering angle for which the wheel slip angle is zero. Finally, the front-wheel longitudinal slip ratio can be recovered as

$$s_{Fx} = \sqrt{s_F^2 - (\tan \alpha_F)^2}.$$

The procedure outlined above can be used to compute the sideslip angle ψ , steering angle δ , and longitudinal slip on the driving wheel s_{Fx} given a desired trim condition (V, ω) . This calculation requires solving three nonlinear equations, namely, (9), (10), and (11). These three scalar equations can be solved in sequence, using efficient numerical procedures. However, for a given pair (V, ω) , there may be no solution, or more than one solutions.

In Figure 1, the region in the (V, a) plane for which at least one solution exists, with $|\psi| \leq 90^\circ$, and $|\delta| \leq 30^\circ$, and $|s_{Fx}| < 1$ (of these constraints, only the one on δ was binding in some cases). This figure reveals an interesting structure in the set of achievable trim conditions for a FWD vehicle.

The bulk of trim conditions can be found in the region (approximately) bounded on the left by the parabola $a = r_k(30^\circ)v^2$, where

$$r_k(\delta) = \sqrt{\ell_R^2 + \left(\frac{\ell_F + \ell_R}{\tan \delta}\right)^2}$$

is the kinematic turn radius, computed assuming that no wheel slipping occurs. In the case at hand, $r_k(30^\circ) = 4.94$ m. This region indicates conditions where wheel slipping is moderate, and does not play a major role in determining the behavior of the car—except possibly for the understeer noticeable at high speeds.

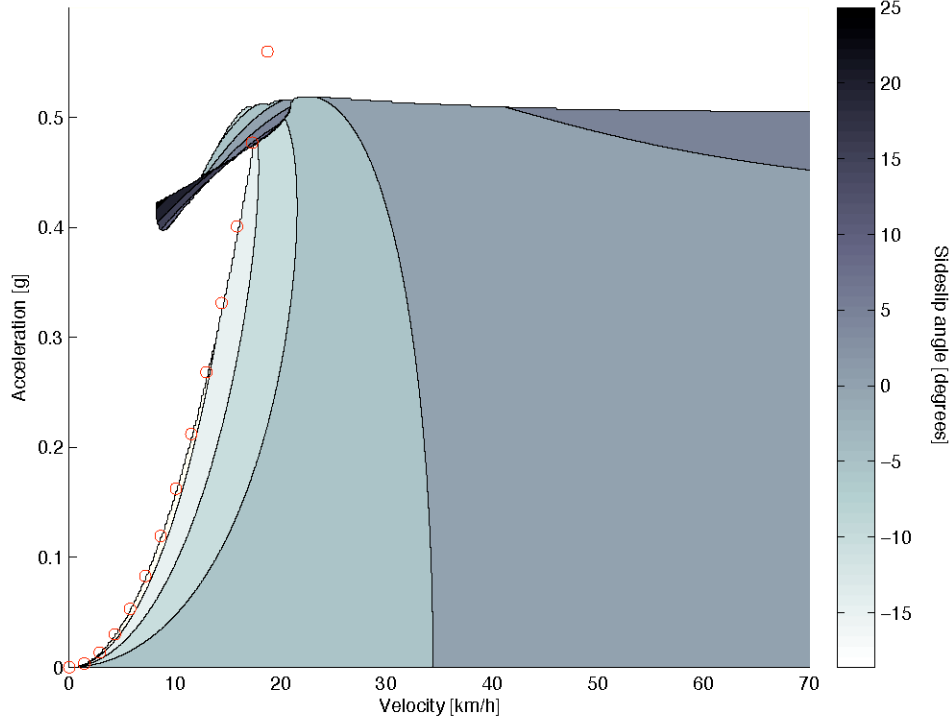


Figure 1: Region of attainable trim conditions for the Front-Wheel Drive vehicle considered in the text. The contour plot shows the sideslip angle ψ (in degrees) required to maintain a trim condition indicated by linear velocity, and centripetal acceleration (such angle is not necessarily unique at all points in the region). The circular markers indicate the kinematic limit on the turning radius.

On the other hand, to the left of the parabola mentioned above, there exist trim conditions in which the vehicle's center of mass travels on a circle with radius smaller than the minimum kinematic turning radius. The sideslip angle is very large, of the order of 25 degrees in our case, and so is the steering angle. This set of trim trajectories corresponds to what are colloquially referred to as “doughnuts,” and the ability of the vehicle to achieve these trajectories can only be modeled taking into account the effects of wheel slipping on the friction forces.

4 Conclusions

In this note, we aimed at complementing the paper under discussion by computing the set of attainable trim trajectories for the half-car model. It is shown that this set is far from trivial, and contains some trajectories corresponding to large sideslip angles, and significant slipping/skidding conditions. These trajectories exceed kinematic limits on the minimum radius of curvature, and on the maximum angular velocity (and hence steering effectiveness) achievable by the car. It is believed that such trajectories can be used, in concert with the techniques developed by Velenis, Tsiotras, and Lu, as further building blocks to construct systematically new classes of “maneuvers,” enabling autonomous cars to design and execute on-line turns at the limit of their performance.

While the real-world implementation of such algorithms on a full-size car is still difficult—especially because of the difficulties in making available to an on-board computer all the sensory cues used by drivers, ranging from visual and auditory data to tactile feedback from the steering wheel—the paper under discussion and this note provide further steps towards a good understanding of the fundamental geometric and dynamic properties of the dynamics of car-like vehicle on loose terrain.

Acknowledgment

Some of the contents of this note, while entirely the author’s responsibility, have been the subject of discussions with K. Iagnemma and R.A. Zachery. Clarifications and feedback from P. Tsiotras and E. Velenis are gratefully acknowledged. The work of the author in this area is partially supported by the US Army Research Office, grant number W911NF-07-1-0499. Any opinions, findings, and conclusions or recommendations expressed in this note are those of the author and do not necessarily reflect the views of the supporting organization.

References

- [1] E. Bakker, L. Nyborg, and H.B. Pacejka. Tyre modelling for use in vehicle dynamics studies. In *Society of Automotive Engineers international congress*, number 870421, Detroit, MI, 1987.
- [2] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas. Symbolic planning and control of robot motion: State of the art and grand challenges. *IEEE Robotics and Automation Magazine*, 14(1):61–70, March 2007.
- [3] E. Frazzoli, M. A. Dahleh, and E. Feron. Maneuver-based motion planning for nonlinear systems with symmetries. *IEEE Trans. on Robotics*, 21(6):1077–1091, December 2005.
- [4] V. Gavrillets, E. Frazzoli, B. Mettler, M. Piedmonte, and E. Feron. Aggressive maneuvering of small autonomous helicopters: A human-centered approach. *International Journal of Robotics Research*, 20(10):795–807, 2001.
- [5] V. Gavrillets, B. Mettler, and E. Feron. Human-inspired control logic for automated maneuvering of miniature helicopter. *AIAA Journal of Guidance, Control, and Dynamics*, 27(5):752–759, 2004.
- [6] E. Velenis, P. Tsiotras, and J. Lu. Optimality properties and driver input parameterization for trail-braking cornering. *European Journal of Control*, 2008.